NO. UMTA-MA-06-0025-75-16.III

AN ASSESSMENT OF THE CRASHWORTHINESS OF EXISTING URBAN RAIL VEHICLES
Volume III: Train-Collision Model Users Manual

D. J. Segal

NOVEMBER 1975
FINAL REPORT

Prepared for

U.S. DEPARTMENT OF TRANSPORTATION
URBAN MASS TRANSPORTATION ADMINISTRATION
Office of Research and Development
Washington DC 20590

NOTICE

This document is disseminated under the sponsorship of the Department of Transportation in the interest of information exchange. The United States Government assumes no liability for its contents or use thereof.

NOTICE

The United States Government does not endorse products or manufacturers. Trade or manufacturers' names appear herein solely because they are considered essential to the object of this report.

Technical Report Documentation Page

1. Report No. UMTA-MA-06-0025-75-16.III	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle AN ASSESSMENT OF THE CRASHWORTHINESS OF EXISTING URBAN RAIL VEHICLES. Volume III: Train-Collision Model Users Manual		5. Report Date November 1975
		6. Performing Organization Code
7. Author(s) D. J. Segal		8. Performing Organization Report No. DOT-TSC-UMTA-75-21.III v.3
9. Performing Organization Name and Address Calspan Corporation* P.O. Box 235 Buffalo NY 14221		10. Work Unit No. (TRAIS) UM604/R6745
		11. Contract or Grant No. DOT-TSC-681
12. Sponsoring Agency Name and Address U.S. Department of Transportation Urban Mass Transportation Administration, Office of Research and Development Washington DC 20590		13. Type of Report and Period Covered Final Report August 1973 - June 1975
		14. Sponsoring Agency Code
15. Supplementary Notes *Under contract to: U.S. Department of Transportation Transportation Systems Center Kendall Square Cambridge MA 02142		

16. Abstract

The crashworthiness of existing urban rail vehicles (passenger cars) and the feasibility of improvements in this area were investigated by Calspan Corporation under contract to the Transportation Systems Center of the U.S. Department of Transportation in its role as systems manager for the Urban Mass Transportation Administration's Urban Rail Supporting Technology Program. Both rail-car structural configurations and impact absorption devices were studied. From this work, recommendations for engineering standards for urban rail vehicles will be developed.

This final report issued under the crashworthiness effort covers:

1. The development of analytical tools to predict passenger threat - environment during collision.
2. Criteria for predicting passenger injury due to train collisions.
3. An application of injury criteria and analytic models to predict passenger injuries resulting from collisions of trains that represent existing construction types.
4. A preliminary investigation of applying impact absorption devices to transit vehicles.
5. A design study of car structural configurations for improved impact energy management.
6. A review of engineering standards for Urban Rail Car Crashworthiness.

The report consists of three volumes:

Volume I: Analyses and Assessments of Vehicles, Chapters 1 Through 7
Volume II: Analyses and Assessments of Vehicles, Chapters 8 Through 12, Appendixes and References
Volume III: Train Collision Model Users Manual.

17. Key Words Rail Car Crashworthiness; Train Collision, Transportation Safety; Crashworhiness, Mathemetical Models		18. Distribution Statement DOCUMENT IS AVAILABLE TO THE PUBLIC THROUGH THE NATIONAL TECHNICAL INFORMATION SERVICE, SPRINGFIELD, VIRGINIA 22161	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 66	22. Price

PREFACE

This Train-Collision Model Users Manual presents documentation of a simplified mathematical model and computer program of the longitudinal dynamics of rail cars involved in collisions and an assessment of the range of subsequent injuries sustained by passengers. The Train-Collision Model, was originally developed in the support of a program of assessment of the crashworthiness of existing urban rail vehicles by the Calspan Corporation for the U.S. Department of Transportation, Transportation Systems Center under Contract No. DOT-TSC-681. Further improvements in the model and documentation were made as reported herein under that contract.

METRIC CONVERSION FACTORS

Approximate Conversions to Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
in	inches	2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km
		AREA		
in^2	square inches	6.5	square centimeters	cm^2
ft^2	square feet	0.09	square meters	m^2
yd^2	square yards	0.8	square meters	m^2
mi^2	square miles	2.6	square kilometers	km^2
	acres	0.4	hectares	ha
		MASS (weight)		
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2000 lb)	0.9	tonnes	t
		VOLUME		
tsp	teaspoons	5	milliliters	ml
Tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.95	liters	l
gal	gallons	3.8	liters	l
ft^3	cubic feet	0.03	cubic meters	m^3
yd^3	cubic yards	0.76	cubic meters	m^3
		TEMPERATURE (exact)		
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C

Approximate Conversions from Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi
		AREA		
cm^2	square centimeters	0.16	square inches	in^2
m^2	square meters	1.2	square yards	yd^2
km^2	square kilometers	0.4	square miles	mi^2
ha	hectares (10,000 m^2)	2.5	acres	
		MASS (weight)		
g	grams	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1000 kg)	1.1	short tons	
		VOLUME		
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m^3	cubic meters	35	cubic feet	ft^3
m^3	cubic meters	1.3	cubic yards	yd^3
		TEMPERATURE (exact)		
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F

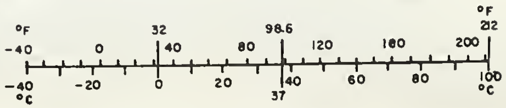


TABLE OF CONTENTS

Section		Page
1.	INTRODUCTION	1-1
2.	MATHEMATICAL-MODEL DESCRIPTION	2-1
	2.1 Rail-Car Model	2-1
	2.2 Calculation of Forces	2-5
	2.3 Rail-Car Integration	2-7
	2.4 Passenger-Injury Assessment	2-9
3.	TRAIN-COLLISION PROGRAM	3-1
	3.1 Input	3-1
	3.2 Output	3-6
	3.3 Sample Runs	3-9
	3.4 Program Glossary	3-18
	3.5 Program Listing	3-20
	3.6 Program Flowcharts	3-31

LIST OF FIGURES

Figure		Page
2-1	Simplified Schematic Diagram of Typical Mode 1 Train Configuration	2-2
2-2	Simplified Schematic Diagram of Typical Mode 2 Train Configuration	2-4
2-3	Force-Deflection Characteristic	2-5
2-4	Force-Calculation Logic	2-6
2-5	Improved Euler Method Integration	2-8
3-1	Train-Collision Program Input	3-2
3-2	Sample Listing of Program Inputs	3-7
3-3	Sample Program Output	3-8
3-4	Inputs For Sample Run 1	3-10
3-5	Mass Displacement - Sample Run 1	3-11
3-6	Inputs For Sample Run 2	3-12
3-7	Car Displacements - Sample Run 2	3-13
3-8	Inputs For Sample Run 3	3-14
3-9	Mass Displacement--Sample Run 3	3-15
3-10	Inputs For Sample Run 4	3-16
3-11	Car Deflections - Sample Run 4	3-17

1. INTRODUCTION

The Train Collision Model is a simplified dynamic simulation of rail-car collisions with automatic estimation of the degree of severity sustained by passengers of each car in a train during second collisions with various types of interior surfaces at different initial distances. The rail cars are modeled as rigid bodies separated by nonlinear, energy-absorbing force deflection characteristics with the first car of a train impacting a rigid barrier. Two modes of operation are possible, the first assuming that an individual force-deflection characteristic acts between each car. For this mode, the deflection of each car (except the first) is assumed to be one-half the relative displacement between the two adjacent car centers of gravity. The second mode of operation allows the individual deflections of each end of each car to be obtained through the use of an inertial mass between the springs representing the car ends. This mode is useful in approximating the different deformations of cars that over-/underride.

Passenger injury-severity distributions are approximated by determining the relative velocity between a mass representing a passenger and the car at times at which the passenger has moved a number of input distances through the car. The relative velocities are subsequently used to calculate a severity index assuming that a passenger in that situation strikes a car fixed object that nulls the relative velocity in a number of input distances at a constant force level.

It must be recognized that the mathematical model and computer simulation described in this report is a very simplified approximation of an extremely complex phenomenon. In this respect, care should be exercised in interpretation of model results as they apply to the real world. While it is expected that generalized trends in overall rail-car crash situations may be properly obtained, results in an absolute sense for specific crash situations should not be indescriminantly interpreted.

2. MATHEMATICAL-MODEL DESCRIPTION

2.1 RAIL-CAR MODEL

A simplified schematic diagram of the mode 1 rail car model is shown in Figure 2-1. The model consists of a barrier with optional movement at a constant velocity, and a number of rail cars (maximum of 20) idealized as rigid bodies separated by nonlinear, energy-absorbing springs. A coordinate system is fixed at the center of gravity of each car and passenger at the instant of impact and by inspection, the acceleration of a car l relative to that coordinate system is given by:

$$\ddot{X}_{c_i} = \frac{1}{M_i}\left(F_{c_{i+1}} - F_{c_i}\right) - a_{c_o}\, g,$$

where a_{c_o} is an arbitrary input deceleration (because of, for example, train braking).

The forces acting between cars arise from deformation of the car ends. Tabular force-deflection characteristics are input to the model, and forces are generated based on the relative positions and velocities of adjoining cars. In this mode of operation, it is assumed that the deflection of the nonlinear springs is shared equally between cars although two force-deflection tables are input so that different characteristics may act between different cars. Thus, for the purpose of determining the force acting between cars i and $i + 1$, the deflection and deflection velocity are

$$\delta_{i+1} = \frac{1}{2}\left(X_{c_{i+1}} - X_{c_i}\right),$$

$$\dot{\delta}_{i+1} = \frac{1}{2}\left(\dot{X}_{c_{i+1}} - X_{c_i}\right).$$

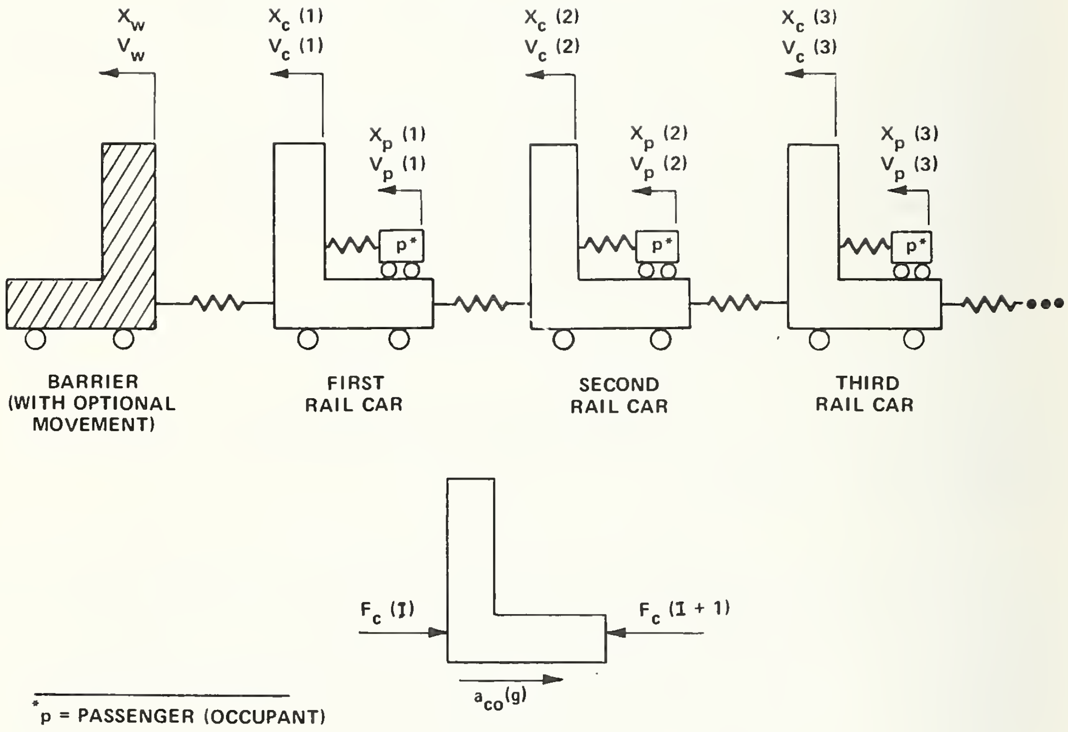


Figure 2-1 Simplified Schematic Diagram of Typical Mode 1 Train Configuration

For the case of the first car of the train,

$$\delta_1 = X_{c_1} - X_w ,$$

$$\dot{\delta}_1 = \dot{X}_{c_1} - V_w .$$

Knowing the deflections and relative velocities of the cars, the forces are determined as described in Section 2.2.

A simplified schematic of the mode 2 rail car model is shown in Figure 2-2. This mode of operation allows independent specification of the force-deflection properties of each adjacent car. An inertial mass is used to separate the cars so that the deflection of each car is calculated independently. This mode is useful in determining the deflections of adjacent car ends when override occurs. In this mode, every other mass is a representation of a rail car, thus a maximum of 10 cars can be simulated. The deflection and deflection velocity of each of the nonlinear springs are given by:

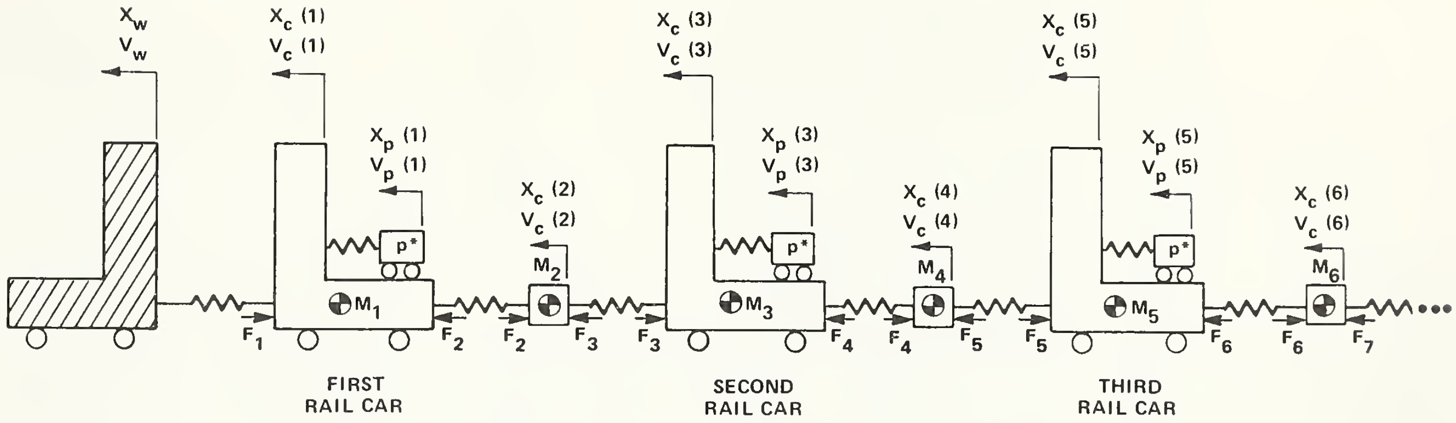
$$\delta_{i+1} = X_{c_{i+1}} - X_{c_i},$$

$$\dot{\delta}_{i+1} = \dot{X}_{c_{i+1}} - X_{c_i},$$

and, for the first spring,

$$\delta_1 = X_{c_1} - X_w,$$

$$\dot{\delta}_1 = \dot{X}_{c_1} - V_w .$$



M_i = WEIGHT/g, i = 1,3,5, . . .

M_j = (WFACTR)*WEIGHT/g, j = 2,4,6, . . .

---

* p = PASSENGER (OCCUPANT)

Figure 2-2 Simplified Schematic Diagram of Typical Mode 2 Train Configuration

2.2 CALCULATION OF FORCES

The calculation of forces acting between rail cars is based on the known deflection and time rate of change of deflection. A general table of force-deflection characteristics is used to interpolate the force level when the deflection is increasing. Unloading and reloading after unloading has occurred takes place along a straight line with a slope of URATE as illustrated in Figure 2-3. If reloading back to F_m, δ_m occurs, calculation of the force along the original curve is resumed.

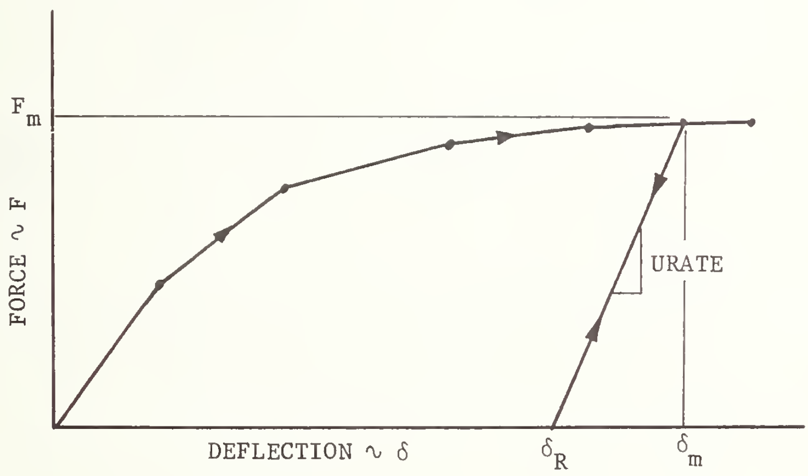


Figure 2-3 Force-Deflection Characteristic

A conceptual flow diagram of the force calculation logic is shown in Figure 2-4.

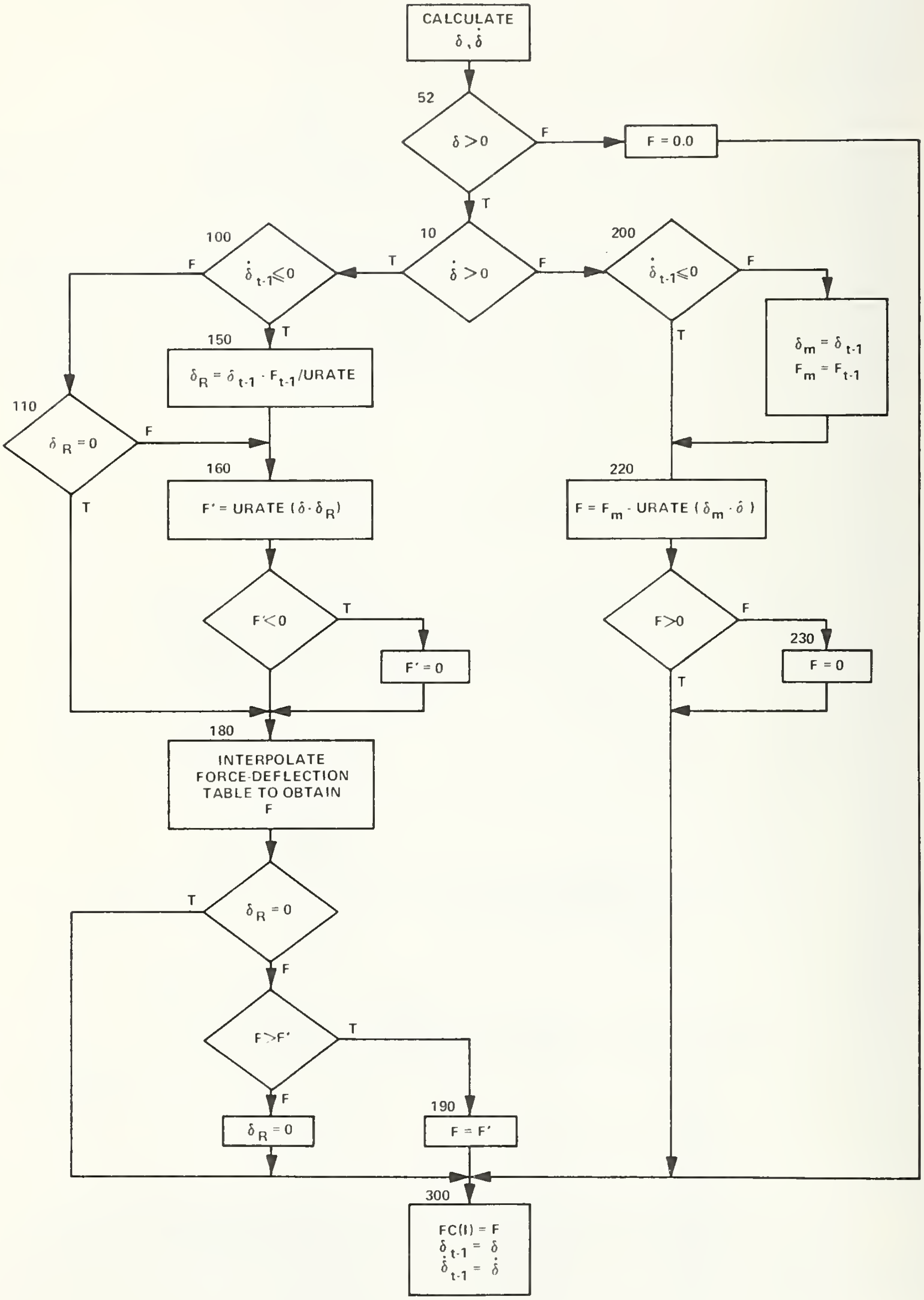


Figure 2-4. Force-Calculation Logic

2.3 RAIL-CAR INTEGRATION

Integration of the differential equations of motion of the rail cars is by the Improved Euler Method as illustrated by Figure 2-5. The procedure is as follows:

a. At the beginning of a time increment, $t = t_m$, determine the acceleration of each car, $a_m^{(1)}$, by calculation of the forces.

b. Assuming a constant acceleration, $a_m^{(1)}$, over the time increment, Δt, approximate the velocity and position of the car at the end of the interval ($t = t_m + \Delta t = t_{m+1}$) by:

$$V_{m+1}^{(0)} = V_m + a_m^{(1)} \Delta t,$$

$$X_{m+1}^{(0)} = X_m + V_m \Delta t + \frac{1}{2} a_m^{(1)} \Delta t^2.$$

c. Determine the acceleration at the end of the interval based on the estimated position and velocity, $a_{m+1}^{(0)}$, by calculation of forces.

d. Assume that the acceleration over the interval is the average of the above calculated accelerations, $\bar{a}_m = 1/2\ (a_m^{(1)} + a_{m+1}^{(o)})$, and compute the revised position and velocity at the end of the interval:

$$V_{m+1}^{(1)} = V_m + \bar{a}_m \Delta t$$

$$= V_m + \frac{1}{2} (a_m^{(1)} + a_{m+1}^{(0)}) \Delta t,$$

$$X_{m+1}^{(1)} = X_m + V_m \Delta t + \frac{1}{2} \bar{a}_m \Delta t^2$$

$$= X_m + V_m \Delta t + \frac{1}{4} (a_m^{(1)} + a_{m+1}^{(0)}) \Delta t^2.$$

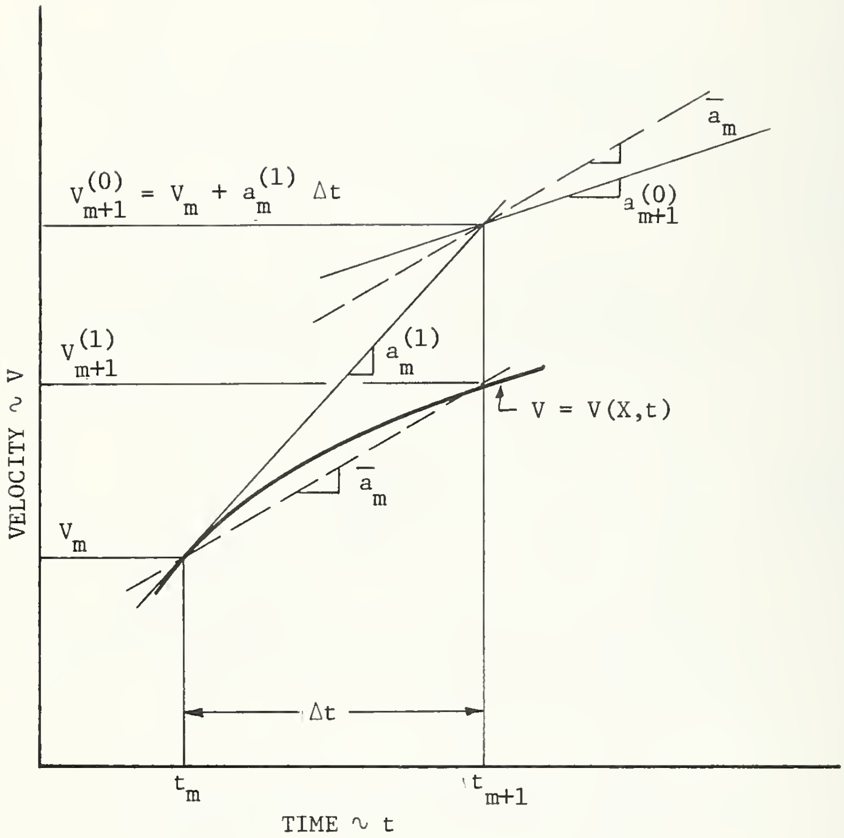


Figure 2-5 Improved Euler Method Integration

2.4 PASSENGER-INJURY ASSESSMENT

The severity of a given crash situation depends in a general sense on the deceleration level acting on the occupant. The deceleration level is a function of the type of surface struck and the relative velocity at impact. Rail cars are, in general, a relatively uncontrolled environment for the occupants after an impact and thus a deterministic approach to modeling occupant dynamics is not at this time warranted.

The approach taken to obtain a general assessment of rail-car crash severity in this model was to determine a range of severity based on the distance the passenger travels before impact and the crush distance of the struck surface. One passenger is simulated in each car and the position and velocity of the passenger are determined assuming that an input constant deceleration acts (simulating for instance, sliding friction). When the position of the passenger relative to the car reaches any one of a number of input spacings (the distance the passenger travels before striking an object), the relative velocity is used to determine an injury-severity index. It is further assumed that the passenger decelerates to a relative velocity of zero in a number of input stopping distances (crush distances) at a constant deceleration level. Note that the effects of ridedown (car deceleration during passenger impact) are neglected in this simplified evaluation.

Given the above assumptions, and assuming a passenger relative velocity of V_r at impact, and a deceleration distance of d, the passenger deceleration level (in g-units) is

$$a_p = \frac{V_r^2}{2dg} \quad .$$

The severity index is:

$$SI = \int_t a_p^{2.5} \, dt = a_p^{2.5}(t),$$

or

$$SI = a_p^{2.5} \frac{V_r}{a_p g} = a_p^{1.5} \frac{v}{g} .$$

3. TRAIN-COLLISION PROGRAM

3.1 INPUT

Refer to Figure 3-1.

Card 1 Format 20A4

TITLE(I), I = 1, 20	Run title. Up to 80 characters of alphanumeric information describing the run	

Card 2 Format 4F10.0, I5

TO	initial-simulation time	sec
TF	final-simulation time	sec
DT	integration-time increment	sec
DTP	output-print interval	sec
MODE	MODE = 0 assumes series springs between cars	-
	MODE = 1 assumes inertial masses acting between cars	

Card 3 Format I5, 5X, 5F10.0

NCARS	number of cars in the train (if MODE = 1, total number of masses)	
WEIGHT	weight of each car	lb
VW	barrier velocity	mph
VTRN	train initial velocity	mph
ACO	train-braking deceleration level	g
APO	passenger deceleration	g

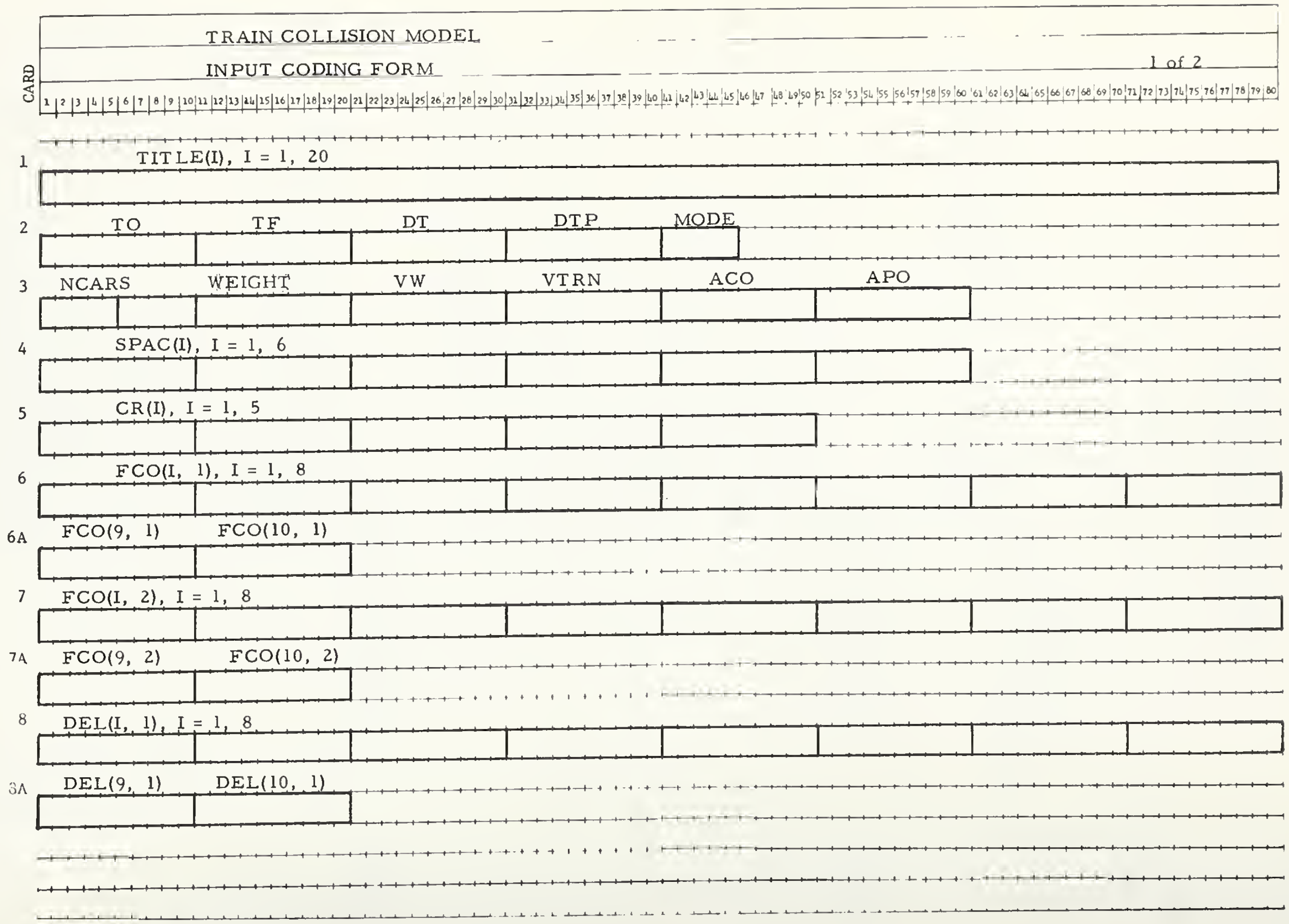


Figure 3-1 Train-Collision Program Input, Sheet 1 of 2

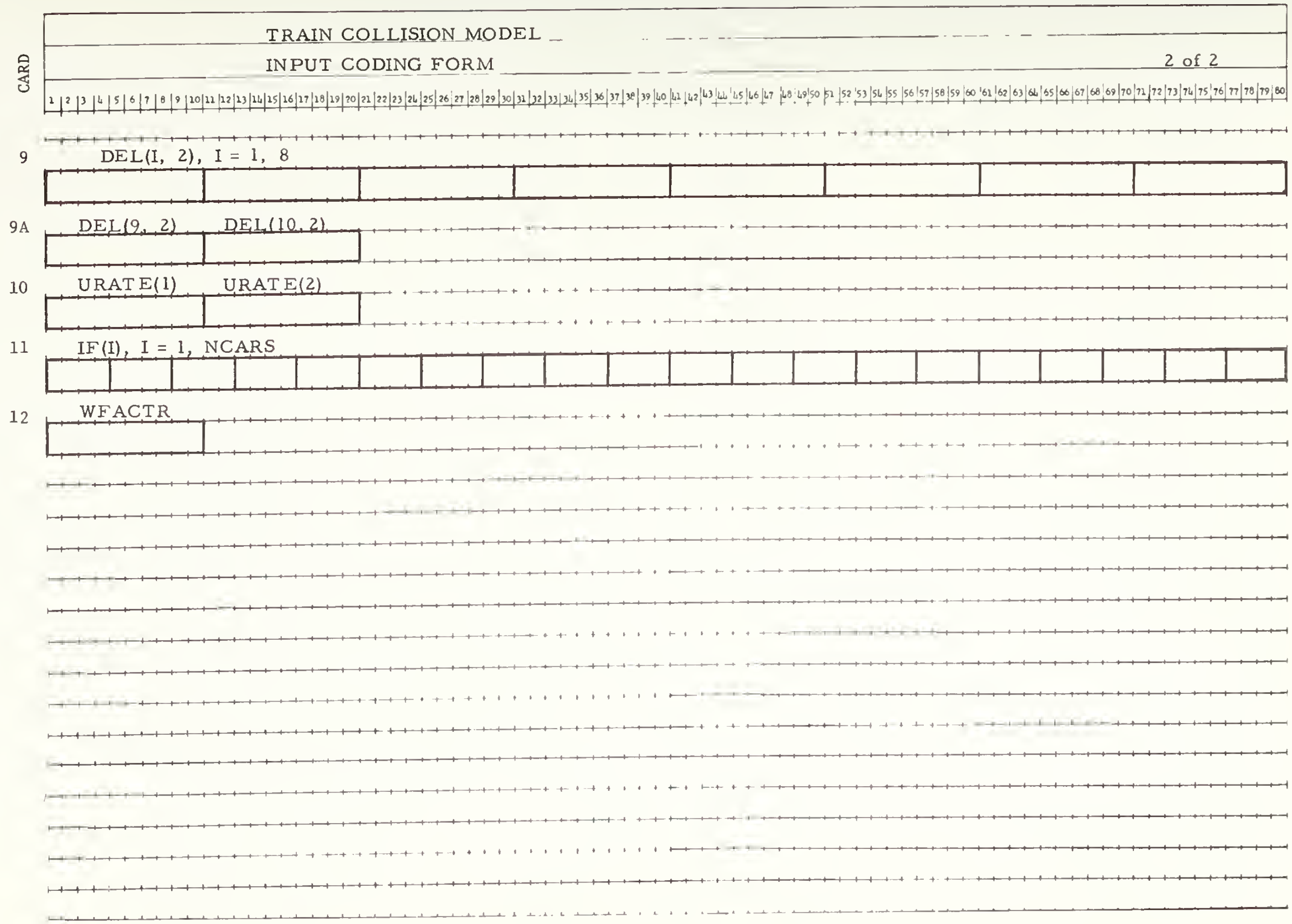


Figure 3-1 Train-Collision Program Input, Sheet 2 of 2

Card 4 Format 6F10.0		
SPAC(I), I = 1,6	passenger-spacing distances before impact	ft
Card 5 Format 6F10.0		
CR(I), I = 1,5	passenger-deceleration distances after impact	in.
Card 6 Format 8G10.0		
FCO(I,1), I = 1,8	force table for first force-deflection characteristic	lb
Card 6A Format 2G10.0		
FCD(I,1), I = 9,10		
Card 7 Format 8G10.0		
FCO(I,1), I = 1,8	force table for second force-deflection characteristic	lb
Card 7A Format 2G10.0		
FCO(I,2), I = 9,10		
Card 8 Format 8G10.0		
DEL(I,1), I = 1,8	deflection table for first force-deflection characteristic	ft
Card 8A Format 2G10.0		
DEL(I,1), I = 9,10		

Card 9 Format 8G10.0

DEL(I,2), I = 1,8	deflection table for second	
Card 9A Format 2G10.0	force-deflection characteristic	ft
DEL(I,2), I = 9,10		

Card 10 Format 2G10.0

URATE(1)	unloading rates for first and	
URATE(2)	second force-deflection	
	characteristics	lb/ft

Card 11 Format 20I4

IF(I), I = 1, NCARS	force-deflection characteristic number (1 or 2) to be used for spring I. Note that spring I acts between masses I-1 and I	

Card 12 Format 1G10.0 (to be read only if MODE = 1)

WFACTR	multiplier used to reduce the weight of masses acting between cars if MODE = 1	
	0 WFACTR 1.0.	

3.2 OUTPUT

Output from the Train-Collision Program includes a listing of program inputs as shown in Figure 3-2, and a time history of dynamic output as shown in Figure 3-3. The dynamic output consists of:

a. The simulated time, in sec

b. The car (or spring) number (note that car number 0 indicates the barrier for which only the position and velocity are output) and for MODE = 1, even-numbered cars represent inertial masses

c. XC, the displacement of each car relative to its position at impact, in ft

d. DEFL, the deflection of the indicated spring. Spring I acts between masses I-1 and I. In the case of MODE = 0, the deflection of spring I is $1/2\ (X_{c_i} - X_{c_{i-1}})$. In the case of MODE = 1, the deflection of spring I is $(X_{c_i} - X_{c_{i-1}})$

e. VC, the velocity of the indicated mass in ft/sec

f. AC, the acceleration of the indicated mass, in g

g. XPC, the relative displacement of the passenger with respect to the car, for the indicated car, in ft

h. VPC, the relative velocity of the passenger with respect to the car for the indicated car, in ft/sec

i. F, the force produced by the indicated spring, in lb.

```
                         TRAIN COLLISION MODEL
                SAMPLE RUN NO. 1   SPRING - MASS SYSTEM

                TO =  0.0     SEC        NO. OF CARS =   1
                TF =  0.10000 SEC          CARWEIGHT =          10.0 LBS
                DT =  0.00200 SEC   BARRIER VELOCITY =           0.0  MPH
          DT PRINT =  0.00020 SEC     TRAIN VELOCITY =          30.00 MPH
              MODE =   0                 TRAIN DECEL. =           0.0  G'S
                                    PASSENGER DECEL. =           0.0  G'S


       TRAIN FORCE-DEFLECTION CHARACTERISTICS
          FCO(1)          DEL(1)          FCO(2)          DEL(2)
           LBS             FT.             LBS             FT.

             0.0            0.0             0.0            0.0
          1226.0            1.00         1226.0            1.00
         12260.0           10.00        12260.0           10.00
             0.0            0.0             0.0            0.0
             0.0            0.0             0.0            0.0
             0.0            0.0             0.0            0.0
             0.0            0.0             0.0            0.0
             0.0            0.0             0.0            0.0
             0.0            0.0             0.0            0.0
             0.0            0.0             0.0            0.0

        UNLOADING RATES: URATE(1)=      1226.00LB/FT          URATE(2)=       1226.00LB/FT


    CAR NO.                                 1
    FORCE-DEFLECTION CHARACTERISTIC NO.     1


       PASSENGER SPACING           PASSENGER DECEL.
       DISTANCE - FT               DISTANCE - IN
            0.10                        1.00
            0.20                        2.00
            0.30                        3.00
            0.40                        4.00
            0.50                        5.00
            0.60
```

Figure 3-2 Sample Listing of Program Inputs

TIME SEC	CAR NO.	XC FT.	DEFL FT.	VC FT/SEC	AC G'S	XPC FT.	VPC FT/SEC	F LB.
0.0	0	0.0		0.0				
	1	0.0	0.0	44.00	0.0	0.0	0.0	.0
0.002	0	0.0		0.0				
	1	0.09	0.09	43.65	-10.79	0.00	0.35	.10789E+03
0.004	0	0.0		0.0				
	1	0.17	0.17	42.62	-21.36	0.00	1.38	.21365E+03
0.006	0	0.0		0.0				
	1	0.26	0.26	40.91	-31.60	0.01	3.09	.31605E+03
0.008	0	0.0		0.0				
	1	0.34	0.34	38.57	-41.35	0.02	5.43	.41347E+03
0.010	0	0.0		0.0				
	1	0.41	0.41	35.61	-50.44	0.03	8.39	.50439E+03
0.012	0	0.0		0.0				
	1	0.48	0.48	32.10	-58.74	0.05	11.90	.58737E+03
0.014	0	0.0		0.0				
	1	0.54	0.54	28.08	-66.11	0.08	15.92	.66111E+03
0.016	0	0.0		0.0				
	1	0.59	0.59	23.62	-72.44	0.11	20.38	.72445E+03
0.018	0	0.0		0.0				
	1	0.63	0.63	18.79	-77.64	0.16	25.21	.77639E+03
0.020	0	0.0		0.0				
	1	0.67	0.67	13.66	-81.61	0.21	30.34	.81612E+03
0.022	0	0.0		0.0				
	1	0.69	0.69	8.32	-84.30	0.28	35.68	.84301E+03
0.024	0	0.0		0.0				
	1	0.70	0.70	2.84	-85.66	0.36	41.16	.85665E+03
0.026	0	0.0		0.0				
	1	0.70	0.70	-2.67	-85.66	0.45	46.67	.85659E+03
0.028	0	0.0		0.0				
	1	0.69	0.69	-8.15	-84.33	0.54	52.15	.84328E+03
0.030	0	0.0		0.0				
	1	0.67	0.67	-13.49	-81.67	0.65	57.49	.81670E+03
0.032	0	0.0		0.0				
	1	0.63	0.63	-18.62	-77.73	0.77	62.62	.77727E+03
0.034	0	0.0		0.0				
	1	0.59	0.59	-23.46	-72.56	0.90	67.46	.72563E+03

Figure 3-3 Sample Program Output

3.3 SAMPLE RUNS

Four sample runs using the Train-Collision Program are presented in this section. The first is a simple linear spring-mass system used to check the accuracy of the integrator. As shown in the program inputs in Figure 3-4, the mass weighs 10 lb and the spring has a rate of 1226 lb/ft. The exact solution to this system yields a natural frequency of 10 cps, and with an initial velocity of 30 mph, the maximum displacement is 0.7003 ft, and maximum acceleration is -85.85 g. The results from the program indicate a maximum displacement of 0.70 ft (Figure 3-5) and maximum acceleration of -85.85 g.

The second and third sample runs compare program results under different modes of operation. Inputs for the second run are shown in Figure 3-6. Two masses, each of which weighs 10 lb, with linear springs, of rates 1226 lb/ft impact a stationary barrier at 30 mph. Displacements of the masses are shown in Figure 3-7.

The third sample run duplicates the conditions of the second (Figure 3-8) except that program mode 1 is used. The two 10-lb masses are separated by a mass of 0.1 lb (WFACTR = 0.01). Results of this run are shown in Figure 3-9. Note that the displacements of the two 10-lb masses are nearly identical with the second run. Although this mode of operation may be useful in determining deflections of individual cars with different force-deflection characteristics, care must be exercised in the choice of integration step size to insure that stability in the integration of the small mass dynamics is maintained.

The fourth sample run is typical of the inputs required for a four-car train. The inputs used are shown in Figure 3-10 and the deflections of the forward ends of each car are shown in Figure 3-11.

```
                         TRAIN COLLISION MODEL
            SAMPLE RUN NO. 1    SPRING - MASS SYSTEM

             TC =  0.0      SEC         NO. OF CARS =   1
             TF =  0.0600   SEC          CARWEIGHT =         10.0 LBS
             DT =  0.00100  SEC    BARRIER VELOCITY =         0.0  MPH
       DT PRINT =  0.0010   SEC      TRAIN VELOCITY =        30.00 MPH
           MODE =  0                   TRAIN DECEL. =         0.0  G'S
                                   PASSENGER DECEL. =         0.0  G'S
```

TRAIN FORCE-DEFLECTION CHARACTERISTICS

FCO(1) LBS	DEL(1) FT.	FCO(2) LBS	DEL(2) FT.
0.0	0.0	0.0	0.0
1226.0	1.00	1226.0	1.00
12260.0	10.00	12260.0	10.00
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0

UNLOADING RATES: URATE(1)= 1226.00LB/FT URATE(2)= 1226.00LB/FT

CAR NO. 1
FORCE-DEFLECTION CHARACTERISTIC NO. 1

PASSENGER SPACING DISTANCE - FT	PASSENGER DECEL. DISTANCE - IN
0.10	1.00
0.20	2.00
0.30	3.00
0.40	4.00
0.50	5.00
0.60	

Figure 3-4 Inputs For Sample Run 1

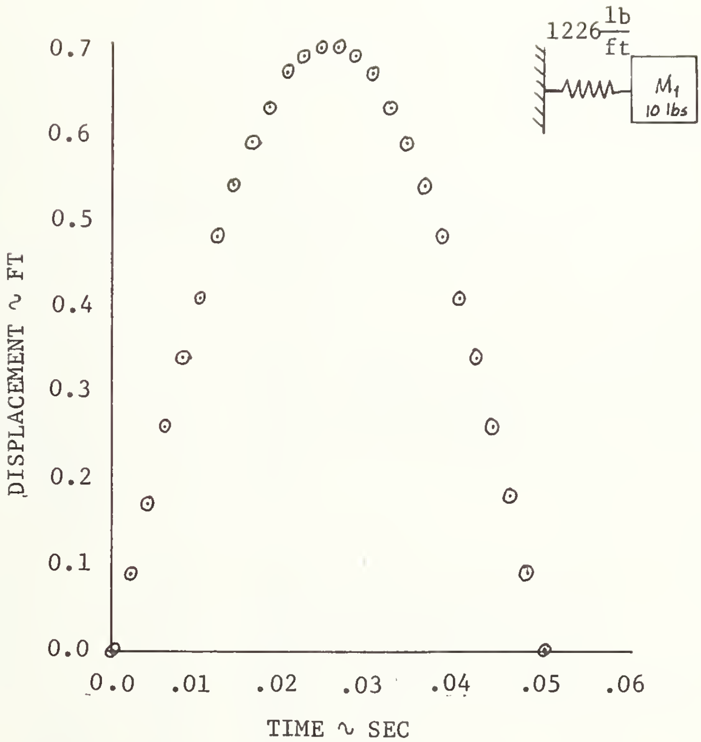


Figure 3-5 Mass Displacement - Sample Run 1

```
                    TRAIN COLLISION MODEL
        SAMPLE RUN NO. 2  TWO MASS SYSTEM  MODE 0

      TO = 0.0      SEC        NO. OF CARS =    2
      TF = 0.1000   SEC          CARWEIGHT =         10.0 LBS
      DT = 0.00200  SEC   BARRIER VELOCITY =          0.0  MPH
DT PRINT = 0.0020   SEC     TRAIN VELOCITY =         30.00 MPH
    MODE = 0                  TRAIN DECEL. =          0.0  G'S
                          PASSENGER DECEL. =          0.0  G'S
```

TRAIN FORCE-DEFLECTION CHARACTERISTICS

FCO(1) LBS	DEL(I) FT.	FCO(2) LBS	DEL(2) FT.
0.0	0.0	0.0	0.0
1226.0	1.00	1226.0	1.00
12260.0	10.00	12260.0	10.00
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0

UNLOADING RATES: URATE(1)= 1226.00LB/FT   URATE(2)= 1226.00LB/FT

CAR NO.	1	2
FORCE-DEFLECTION CHARACTERISTIC NO.	1	1

PASSENGER SPACING DISTANCE - FT	PASSENGER DECEL. DISTANCE - IN
0.10	1.00
0.20	2.00
0.30	3.00
0.40	4.00
0.50	5.00
0.60	

Figure 3-6 Inputs For Sample Run 2

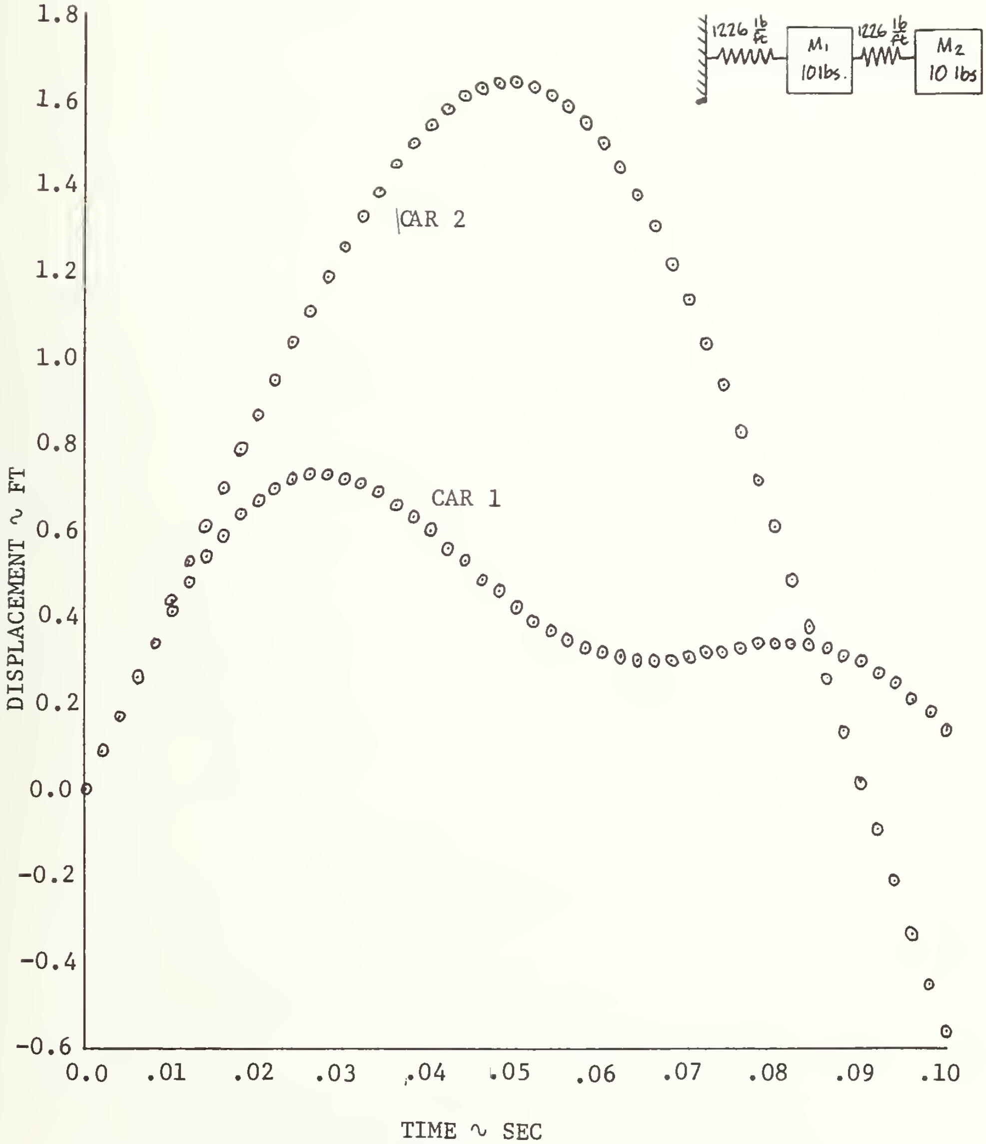


Figure 3-7 Car Displacements - Sample Run 2

```
                         TRAIN COLLISION MODEL
                SAMPLE RUN NO. 3   3 MASS MODE 1

          TC =  0.0     SEC          NO. OF CARS =    3
          TF =  0.1000  SEC           CARWEIGHT =          10.0 LBS
          DT =  0.00200 SEC     BARRIER VELOCITY =          0.0  MPH
    DT PRINT =  0.0020  SEC       TRAIN VELOCITY =         30.00 MPH
        MODE =  1                   TRAIN DECEL. =          0.0  G'S
                                PASSENGER DECEL. =          0.0  G'S

   TRAIN FORCE-DEFLECTION CHARACTERISTICS
       FCO(1)          DEL(1)          FCO(2)          DEL(2)
        LBS              FT.             LBS              FT.

         0.0            0.0             0.0             0.0
      1226.0            1.00         1226.0             1.00
     12260.0           10.00        12260.0            10.00
         0.0            0.0             0.0             0.0
         0.0            0.0             0.0             0.0
         0.0            0.0             0.0             0.0
         0.0            0.0             0.0             0.0
         0.0            0.0             0.0             0.0
         0.0            0.0             0.0             0.0
         0.0            0.0             0.0             0.0

   UNLOADING RATES: URATE(1)=     1226.00LB/FT       URATE(2)=      1226.00LB/FT

 CAR NO.                                  1     2     3
 FORCE-DEFLECTION CHARACTERISTIC NO.      1     1     1

   PASSENGER SPACING             PASSENGER DECEL.
   DISTANCE - FT                 DISTANCE - IN
        0.10                          1.00
        0.20                          2.00
        0.30                          3.00
        0.40                          4.00
        0.50                          5.00
        0.60

 WFACTR =   0.0100
```

Figure 3-8   Inputs For Sample Run 3

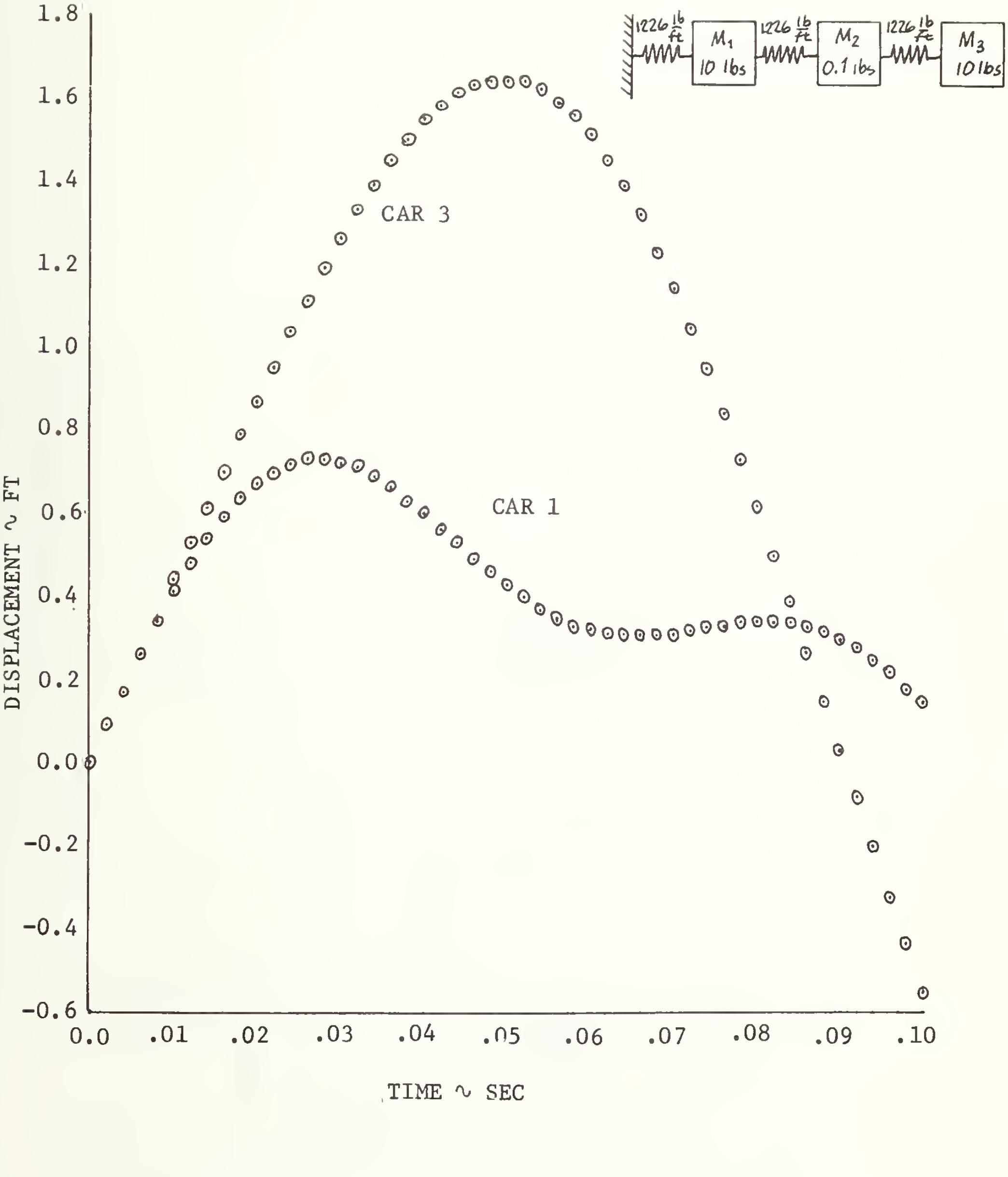


Figure 3-9 Mass Displacement--Sample Run 3

TRAIN COLLISION MODEL

SAMPLE RUN NO. 4 4 CAR TRAIN

T0 =	0.0 SEC	NO. OF CARS =	4
TF =	0.5000 SEC	CARWEIGHT =	60000.0 LBS
DT =	0.00100 SEC	BARRIER VELOCITY =	0.0 MPH
DT PRINT =	0.0100 SEC	TRAIN VELOCITY =	20.00 MPH
MODE =	0	TRAIN DECEL. =	0.0 G'S
		PASSENGER DECEL. =	0.0 G'S

TRAIN FORCE-DEFLECTION CHARACTERISTICS

FCO(1) LBS	DEL(1) FT.	FCO(2) LBS	DEL(2) FT.
0.0	0.0	0.0	0.0
165000.0	0.25	165000.0	0.25
0.0	0.26	0.0	0.26
0.0	0.42	0.0	0.42
720000.0	0.50	720000.0	0.50
720000.0	4.58	720000.0	4.58
1170000.0	4.67	1170000.0	4.67
1170000.0	66.67	1170000.0	66.67
1170000.0	70.00	1170000.0	70.00
1170000.0	75.00	1170000.0	75.00

UNLOADING RATES: URATE(1)= 8640000.00LB/FT URATE(2)= 8640000.00LB/FT

CAR NO.	1	2	3	4
FORCE-DEFLECTION CHARACTERISTIC NO.	1	1	1	1

PASSENGER SPACING DISTANCE - FT	PASSENGER DECEL. DISTANCE - IN
2.00	0.25
4.00	1.00
6.00	2.00
9.00	4.00
12.00	12.00
24.00	

Figure 3-10 Inputs For Sample Run 4

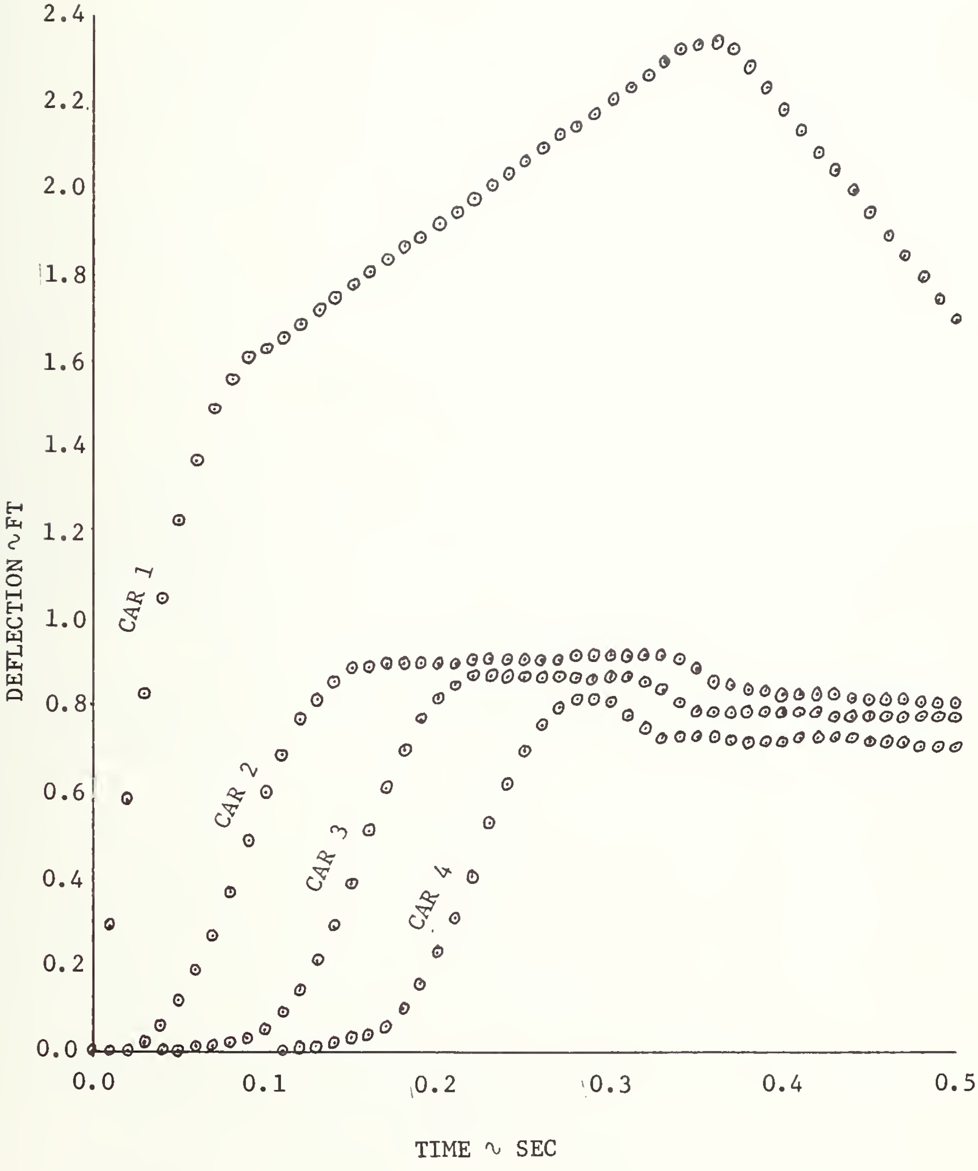


Figure 3-11 Car Deflections - Sample Run 4

3.4 PROGRAM GLOSSARY

Program Symbol	Definition
AC	car or mass acceleration
ACO	car-braking deceleration
AIMP	passenger impact deceleration
AP	passenger acceleration
APO	passenger-braking deceleration
CR	passenger crush distance
DEF	car deflection
DEFV	car-deflection velocity
DEL	deflection table
DM	deflection start of unloading
DP	previous value of DEF (at T-DT)
DR	zero force intercept of reloading curve
DT	integration increment
DTP	printout interval
DVP	previous value of DEFV (at T-DT)
FC	intermass force
FCO	force table
FM	force level at start of unloading
G	acceleration of gravity, 32.2 ft/sec^2
IF	force-deflection number indicator
IMP	passenger-impact indicator
MODE	indicator for simulation mode of operation

Program Symbol	Definition
N, NCARS	number of cars or masses
SI	passenger severity index table
SPAC	passenger spacing before impact
T	simulated time
TF	end time
TITLE	run title
TO	initial time
URATE	unloading rate
VC	car or mass velocity
VIMP	passenger relative velocity at impact
VP	passenger velocity
VTRN	train initial velocity
VW	barrier velocity
WEIGHT	car weight
WFACTR	factor for reducing weight of inter-car inertial masses
WGT	car or mass weight
XC	car position
XP	passenger position
XW	barrier position

3.5 PROGRAM LISTING

```
C          ***  RAIL CAR COLLISION MODEL  *****
C****
      COMMON IMP(20),WGT(20),FC(21),XC(20),VC(20),AC(20),XP(20),VP(20),
     1       AP(20),VIMP(20,6),DVP(20),DP(20),DM(20),FM(20),URATE(2),
     2       DR(20),DEL(10,2),FCO(10,2),IF(20),TITLE(20),TO,NCARS,TF,
     3       WEIGHT,DT,VW,DTP,VTRN,MODE,ACO,APO,SPAC(6),CR(5),WFACTR,
     4       XW    ,T,G,N,AIMP(20,6,5),SI(20,6,5)
C****
C         CLEAR COMMON STORAGE TO ZERO
C****
    1 CALL CLEAR(IMP(1),SI(20,6,5))
C****
C          INITIALIZE OCCUPANT CONTACT INDICATOR
C****
    4 DO 2 I=1,20
    2 IMP(I) = 1
C****
C      OBTAIN INPUT AND INITIALIZE
C****
    3 CALL INPUT
      TPRNT = TO+DTP
      G = 32.2
      FPS = 5280./3600.
      VWALL = VW   *FPS
      VTRN = VTRN*FPS
      N = NCARS
      N1 = 1
      DO 31 I=1,N
   31 WGT(I) = WEIGHT
C****
C        MODE=1 INDICATES THAT EVERY OTHER CAR IS AN INERTIAL MASS
C          OF WEIGHT  WFACTR*WGT BETWEEN CARS SO THAT DIFFERENT
C          FORCE-DEFLECTION CHARACTERISTICS CAN BE USED ON
C          ADJACENT CARS
C****
C****
C       INITIALIZATION
C****
      IF(MODE.NE.1) GO TO 28
      DO 29 I=2,N,2
   29 WGT(I) = WGT(I)*WFACTR
   28 XW = 0.0
      VW = VWALL
      T = TO
      DO 32 I=1,N
      FC(I) = 0.0
      XC(I) = 0.0
      VC(I) = VTRN
      AC(I) = ACO*G
      XP(I) = 0.0
      VP(I) = VTRN
   32 AP(I) = 0.0
      CALL OUTPUT(1)
C****
C          BEGIN INTEGRATION LOOP.  INTEGRATION BY MODIFIED EULER METHOD
```

```
C****
   10 XW = VW*T
C****
C      EVALUATE ACCELERATION AT T
C****
      CALL FORCE
      DO 707 I=1,N
      ACT = 0.0
      IF(ABS(VC(I)).GT.0.01) ACT = ACO*VC(I)/ABS(VC(I))
      XDD = (FC(I+1)-FC(I))*G/WGT(I)+ACT
C****
C      PROJECT POSITION AND VELOCITY AT T+DT
C****
      XD = VC(I)+XDD*DT
      X = XC(I)+VC(I)*DT+.5*XDD*DT*DT
      XC(I) = X
      VC(I) = XD
  707 AC(I) = XDD
      T = T+DT
      XW = VW*T
C****
C      EVALUATE ACCELERATION AT T+DT
C****
      CALL FORCE
      DO 706 I=1,N
      ACT = 0.0
      IF(ABS(VC(I)).GT.0.01) ACT = ACO*VC(I)/ABS(VC(I))
      XDD = (FC(I+1)-FC(I))*G/WGT(I)+ACT
C****
C      REVISE POSITION AND VELOCITY AT T+DT
C****
      XD1 = VC(I)-AC(I)*DT
      XD = XD1+.5*(AC(I)+XDD)*DT
      X1 = XC(I)-XD1*DT-.5*AC(I)*DT*DT
      X = X1+XD1*DT+.25*(AC(I)+XDD)*DT*DT
      XC(I) = X
      VC(I) = XD
  706 AC(I) = XDD
      N1 = 1
      IF(MODE.EQ.1) N1 = 2
C****
C         INTEGRATE POSITION AND VELOCITY OF PASSENGERS IN EACH CAR
C****
      DO 40 I=1,N,N1
      VDIF = VC(I)-VP(I)
      AP(I) = 0.0
      IF(VDIF.NE.0.0) AP(I) = APO*VDIF/ABS(VDIF)
      VP(I) = VP(I)+AP(I)*DT
      XP(I) = XP(I)+VP(I)*DT+.5*AP(I)*DT*DT
C****
C      CALCULATE RELATIVE POSITION AND VELOCITY OF PASSENGER
C***
      XPR = XP(I)-XC(I)
      VPR = VP(I)-VC(I)
      IS = IMP(I)
```

```
      IF(IS.GE.7) GO TO 40
C****
C          IF PASSENGER HAS MOVED INPUT DISTANCES OF SPAC RELATIVE TO
C          CAR, LOAD RELATIVE VELOCITY INTO VIMP FOR SUBSEQUENT SEVERITY
C          INDEX CALCULATIONS
C      SPAC MUST BE IN INCREASING ORDER
C****
      IF(XPR.LT.SPAC(IS)) GO TO 40
      IMP(I) = IS+1
      VIMP(I,IS) = VPR
   40 CONTINUE
      IF(T.GT.TF) GO TO 900
C****
C          OUTPUT IF PRINT INTERVAL HAS BEEN REACHED
C****
      IF(TPRNT.GT.T+.1*DT) GO TO 10
      CALL OUTPUT(2)
      TPRNT = TPRNT+DTP
      GO TO 10
C****
C          AT END OF RUN , CALCULATE SEVERITY INDEX
C****
  900 CALL SIC
      GO TO 4
      END
```

```
      SUBROUTINE OUTPUT(IND)
C****
C           PRINTS DYNAMIC OUTPUT FROM TRAIN COLLISION MODEL
C****
      COMMON IMP(20),WGT(20),FC(21),XC(20),VC(20),AC(20),XP(20),VP(20),
     1       AP(20),VIMP(20,6),DVP(20),DP(20),DM(20),FM(20),URATE(2),
     2       DR(20),DEL(10,2),FCO(10,2),IF(20),TITLE(20),TO,NCARS,TF,
     3       WEIGHT,DT,VW,DTP,VTRN,MODE,ACO,APO,SPAC(6),CR(5),WFACTR,
     4       XW   ,T,G,N,AIMP(20,6,5),SI(20,6,5)
C****
C      FOR IND=1, CALCULATE IPRNT FOR OUTPUT PAGING
C****
      IF(IND.GT.1) GO TO 10
      IPRNT = 55/(NCARS+2)
    5 LL = 0
C****
C      OUTPUT PAGE HEADINGS
C****
      WRITE(6,2000)
 2000 FORMAT(1H1,1X,4HTIME,6X,3HCAR,5X,2HXC,8X,4HDEFL,6X,2HVC,
     1     8X,2HAC,8X,3HXPC,7X,3HVPC,7X,1HF    /
     2     2X,3HSEC,7X,3HNO.,5X,3HFT.,7X,3HFT.,7X,6HFT/SEC,4X,3HG'S,
     3     7X,3HFT., 7X,6HFT/SEC,4X,3HLB.    //   )
   10 LL = LL+1
C****
C      LL IS A COUNTER FOR PAGING
C****
      IF(LL.GT.IPRNT) GO TO 5
      I = 0
C****
C      OUTPUT FOR BARRIER
C****
      WRITE(6,2001)  T,I,XW,VW
 2001 FORMAT(1H0,F8.3,2X,I2,2X,F8.2,12X,F8.2)
      FCTR = .5
      IF(MODE.EQ.1) FCTR = 1.0
      DO 34 I= 1,N
      FI = FC(I)
      XI = XC(I)
      VI = VC(I)
      AI = AC(I)/G
      XPC =  XP(I)-XI
      VPC = VP(I)-VI
      IF(I.GT.1) GO TO 32
      DI = XI-XW
      GO TO 31
   32 DI = (XI-XC(I-1))*FCTR
C****
C      OUTPUT DYNAMIC DATA FOR MASSES AND SPRINGS
C****
   31 WRITE(6,2002) I,XI,DI,VI,AI,XPC,VPC,FI
 2002 FORMAT(1H ,10X,I2,6(2X,F8.2),2X,E10.5)
   34 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE FORCE
C****
C          CALCULATES FORCE ACTING BETWEEN CARS AS A FUNCTION OF DISTANCE
C****
      COMMON IMP(20),WGT(20),FC(21),XC(20),VC(20),AC(20),XP(20),VP(20),
     1       AP(20),VIMP(20,6),DVP(20),DP(20),DM(20),FM(20),URATE(2),
     2       DR(20),DEL(10,2),FCO(10,2),IF(20),TITLE(20),TC,NCARS,TF,
     3       WEIGHT,DT,VW,DTP,VTRN,MODE,ACO,APO,SPAC(6),CR(5),WFACTR,
     4       XW    ,T,G,N,AIMP(20,6,5),SI(20,6,5)
      DO 50 I=1,N
C****
C       NF INDICATES THE APPROPRIATE FORCE-DEFLECTION CHARACTERISTIC
C       FOR SPRING I
C****
      NF = IF(I)
      IF(I.NE.1) GO TO 51
      DEF = XC(1)-XW
      DEFV = VC(1)-VW
      GO TO 52
   51 FACTOR = 1.0
C****
C          IF MODE=0, FORCE DEFLECTION CHARACTERISTICS BETWEEN CARS
C           ARE SERIES COMBINATIONS OF THE INDIVIDUAL CAR PROPERTIES.
C           THE DEFLECTION OF EACH CAR IS ASSUMED TO BE .5*THE TOTAL.
C****
      IF(MODE.EQ.0) FACTOR = .5
      DEF = (XC(I)-XC(I-1))*FACTOR
      DEFV = VC(I)-VC(I-1)
   52 IF(DEF.GT.0.) GO TO 10
      F = 0.0
      GO TO 300
   10 IF(DEFV.GT.0.) GO TO 100
  200 IF(DVP(I).LT.0.) GO TO 220
C****
C          SET UNLOADING CURVE
C****
      DM(I) = DEF
      FM(I) = FC(I)
C****
C          CALCULATE UNLOADING FORCE
C****
  220 F = FM(I)-URATE(NF)*(DM(I)-DEF)
      IF(F.GE.0.0) GO TO 300
  230 F = 0.0
      GO TO 300
  100 IF(DVP(I).LE.0.0) GO TO 150
  110 IF(DR(I).EQ.0.0) GO TO 180
      GO TO 160
C****
C           SET RELOADING CURVE
C     DR IS THE ZERO FORCE INTERCEPT FOR RELOADING
C****
  150 DR(I) = DP(I)-FC(I)/URATE(NF)
C****
C          CALCULATE FORCE FOR RELOADING ALONG UNLOADING CURVE
```

```
C****
  160 FP = URATE(NF)*(DEF-DR(I))
      IF(FP.LT.0.0) FP = 0.0
C****
C          INTERPOLATE FORCE ON LOADING CURVE
C****
  180 DO 185 K=2,10
      IF(DEF.GT.DEL(K,NF)) GO TO 185
      K1 = K-1
      F = FCO(K1,NF)+(FCO(K,NF)-FCO(K1,NF))*(DEF-DEL(K1,NF))
     1     /(DEL(K,NF)-DEL(K1,NF))
      GO TO 186
  185 CONTINUE
      F = FCO(9,NF)+(FCO(10,NF)-FCO(9,NF))*(DEF-DEL(9,NF))/
     1        (DEL(10,NF)-DEL(9,NF))
  186 CONTINUE
      IF(DR(I).EQ.0.0) GO TO 300
C****
C      IF RELOADING, MAKE SURE FORCE LEVEL ON RELOADING CURVE DOES NOT
C      EXCEED FORCE ON ORIGINAL LOADING TABLE
C****
      IF(F.GT.FP) GO TO 190
      DR(I) = 0.0
      GO TO 300
  190 F = FP
  300 FC(I) = F
C****
C      SET PREVIOS VALUES FOR NEXT TIME
C****
      DP(I) = DEF
      DVP(I) = DEFV
   50 CONTINUE
      FC(N+1) = 0.0
      RETURN
      END
```

```
      SUBROUTINE SIC
C****
C          CALCULATES SEVERITY INDEX MATRIX FOR EACH CAR FOR PASSENGER
C          DISTANCES OF SPAC BEFORE IMPACT AND STOPPING DISTANCES
C         OF CR.  NOTE THAT THE RELATIVE VELOCITY OF THE PASSENGER
C          W/TO THE CAR AT SPACING DISTANCES SPAC ARE VIMP AS CALCULATED
C          IN MAIN
C***
      COMMON IMP(20),WGT(20),FC(21),XC(20),VC(20),AC(20),XP(20),VP(20),
     1       AP(20),VIMP(20,6),DVP(20),DP(20),DM(20),FM(20),URATE(2),
     2       DR(20),DEL(10,2),FCO(10,2),IF(20),TITLE(20),TO,NCARS,TF,
     3       WEIGHT,DT,VW,DTP,VTRN,MODE,ACO,APO,SPAC(6),CR(5),WFACTR,
     4       XW    ,T,G,N,AIMP(20,6,5),SI(20,6,5)
      N1 = 1
      IF(MODE.EQ.1) N1 = 2
      DO 800 I=1,N,N1
      DO 800 K=1,6
      DO 800 M=1,5
C****
C          AIMP IS CONSTANT ACCELERATION FROM VELOCITY VIMP IN DISTANCE CR.
C****
      AIMP(I,K,M) = 6.*VIMP(I,K)**2/(CR(M)*G)
C****
C           SEVERITY INDEX IS INTEGRAL OF ACCELERATION**2.5 OVER TIME
C****
      SI(I,K,M) = AIMP(I,K,M)**1.5*VIMP(I,K)/G
  800 CONTINUE
C****
C       OUTPUT SEVERITY INDEX TABULATION AS A FUNCTION OF PASSENGER
C       SPACING AND CRUSH DISTANCE FOR EACH CAR
C****
      WRITE(6,3000)
 3000 FORMAT(1H1,// 40X,14HSEVERITY INDEX  )
      DO 950 I=1,N,N1
      WRITE(6,3001) I,(SPAC(K),K=1,6)
 3001 FORMAT(1H0,10X,3HCAR,I3 / 3X,5HCRUSH,22X,12HSPACING - FT    /
     1         4X,3HIN.,3X,6F10.2  )
      DO 950 K=1,5
      WRITE(6,3002) CR(K),(SI(I,J,K),J=1,6)
 3002 FORMAT(1H0,F7.2,2X,6F10.1  / (1X,F7.2,2X,6F10.1) )
  950 CONTINUE
      IF(MODE.EQ.1) WRITE(6,3003)
 3003 FORMAT(1H0,//47H NOTE: MODE=1, THEREFORE PASSENGERS ASSOCIATED
     1     20HWITH EVERY OTHER CAR       )
      RETURN
      END
```

```
      SUBROUTINE INPUT
C****
C         READS TRAIN COLLISION MODEL INPUT FROM CARDS AND PRINTS INPUT DATA
C****
      COMMON IMP(20),WGT(20),FC(21),XC(20),VC(20),AC(20),XP(20),VP(20),
     1       AP(20),VIMP(20,6),DVP(20),DP(20),DM(20),FM(20),URATE(2),
     2       DR(20),DEL(10,2),FCO(10,2),IF(20),TITLE(20),TO,NCARS,TF,
     3       WEIGHT,DT,VW,DTP,VTRN,MODE,ACO,APO,SPAC(6),CR(5),WFACTR,
     4       XW   ,T,G,N,AIMP(20,6,5),SI(20,6,5)
C****
C      READ INPUT
C****
C****
C      RUN TITLE CARD, 80 CHARACTERS
C****
      READ(5,99) (TITLE(I),I=1,20)
   99 FORMAT(20A4)
C****
C      PROGRAM CONTROL
C****
      READ(5,98) TO,TF,DT,DTP,MODE
   98 FORMAT(4F10.0,I5)
      READ(5,97) NCARS,WEIGHT,VW,VTRN,ACO,APO
   97 FORMAT(I5,5X,5F10.0)
      READ(5,96) (SPAC(I),I=1,6)
      READ(5,96) (CR(I),I=1,5)
   96 FORMAT(6F10.0)
C****
C      EACH FCO AND DEL REQUIRES 2 CARDS WITH 2 VALUES ON SECOND CARD
C      BLANK CARD MUST BE FURNISHED IF NO DATA FOR LAST TWO ENTRIES.
C****
      READ(5,95) (FCO(I,1),I=1,10)
      READ(5,95) (FCO(I,2),I=1,10)
      READ(5,95) (DEL(I,1),I=1,10)
      READ(5,95) (DEL(I,2),I=1,10)
      READ(5,95) URATE(1),URATE(2)
   95 FORMAT(8G10.0)
C****
C      IF INDICATES WHICH FCO-DEL TABLE USED FOR EACH SPRING.  MUST
C      BE 1 OR 2
C****
      READ(5,94) (IF(I),I=1,NCARS)
   94 FORMAT(20I4)
C****
C      WFACTR READ ONLY IF MODE = 1
C****
      IF(MODE.EQ.1) READ(5,95) WFACTR
C****
C      PRINT INPUT
C****
      WRITE(6,1000) (TITLE(I),I=1,20)
 1000 FORMAT(1H1,40X,21HTRAIN COLLISION MODEL    /  20X,20A4)
      WRITE(6,1001) TO,NCARS,TF,WEIGHT,DT,VW,DTP,VTRN,MODE,ACO,APO
      WRITE(6,1002) (FCO(I,1),DEL(I,1),FCO(I,2),DEL(I,2),I=1,10)
      WRITE(6,1003) URATE(1),URATE(2)
```

```
      WRITE(6,1004) (I,I=1,NCARS)
      WRITE(6,1005) (IF(I),I=1,NCARS)
      WRITE(6,1006) (SPAC(I),CR(I),I=1,5),SPAC(6)
      IF(MODE.EQ.1) WRITE(6,1007) WFACTR
 1007 FORMAT(1H0,  // 9H WFACTR =,F10.4)
 1001 FORMAT(1H0,25X,4HT0 =,F8.4,4H SEC,8X,13HNO. OF CARS =,I4, /
     1   26X,4HTF =,F8.4,4H SEC,9X,11HCARWEIGHT =,F12.1,4H LBS /
     2   26X,4HDT =,F8.5,4H SEC,3X,18HBARRIER VELOCITY =,F12.2,4H MPH/
     3   20X,10HDT PRINT =,F8.4,4H SEC,5X,16HTRAIN VELOCITY =,F12.2,
     4   4H MPH /
     5  24X,6HMODE =,I4,15X,14HTRAIN DECEL. =,F12.2,4H G'S  /
     6   45X,18HPASSENGER DECEL. =,F12.3,4H G'S   )
 1002 FORMAT(1H0, // 10X,38HTRAIN FORCE-DEFLECTION CHARACTERISTICS   /
     1   13X,6HFCO(1),10X,6HDEL(1),10X,6HFCO(2),10X,6HDEL(2)   /
     2   14X,3HLBS,13X,3HFT.,13X,3HLBS,13X,3HFT.  //
     3   (10X,F10.1,8X,F7.2,7X,F10.1,8X,F7.2)    )
 1003 FORMAT(1H0,10X,26HUNLOADING RATES: URATE(1)=,F14.2,5HLB/FT,
     2   10X,9HURATE(2)=,F14.2,5HLB/FT  /)
 1004 FORMAT(1H0, // 5X,7HCAR NO.,28X,15I5 )
 1005 FORMAT(1H ,4X,35HFORCE-DEFLECTION CHARACTERISTIC NO.  ,15I5  )
 1006 FORMAT(1H0, // 10X,17HPASSENGER SPACING,13X,16HPASSENGER DECEL. /
     1   10X,13HDISTANCE - FT,17X,13HDISTANCE - IN  /
     2   (13X,F8.2,22X,F8.2)    )
      RETURN
      END
```

```
      SUBROUTINE CLEAR(A,B)
C       CLEARS (SETS TO ZERO) A BLOCK OF STORAGE IDENTIFIED BY
C       THE ADDRESSES OF THE TWO ARGUMENTS
C          CALL CLEAR(P,Q) WILL CAUSE ALL BYTES TO BE SET TO
C          ZERO FROM ADDRESS P THROUGH THE FULL-WORD ADDRESS AT Q
      DIMENSION A(1),B(1)
      B(1) = 1.0
      I = 0
   10 IF(B(1).EQ.0.0) RETURN
      I = I+1
      A(I) = 0.0
      GO TO 10
      END
```

3.6 PROGRAM FLOWCHARTS

CHART TITLE - PROCEDURES

*** RAIL CAR COLLISION MODEL *****

```
/ 1 /

CLEAR COMMON STORAGE TO ZERO
01
I/O  CLEAR (IMP(1),SI(2C,6,5))

INITIALIZE OCCUPANT CONTACT INDICATO-
04.12--->
NOTE 02
BEGIN DO LOOP 2 I = 1, 20
2   03
IMP(I) = 1
04
END OF DO LOOP?   NO / YES

OBTAIN INPUT AND INITIALIZE
3   05
I/O  INPUT
06
TPRNT = TO + DTP
G = 32.2
FPS = 5280./3600.
VWALL = VW*FPS
07
VTRN = VTRN*FPS
N = NCARS
NI = 1
NOTE 08
BEGIN DO LOOP 31 I = 1, N
31   09
WGT(I) = WEIGHT

10
END OF DO LOOP?   NO / YES

MODE=1 INDICATES THAT EVERY OTHER CAR IS AN
INERTIAL MASS OF WEIGHT WFACTR*WGT BETWEEN CARS
SO THAT DIFFERENT FORCE-DEFLECTION
CHARACTERISTICS CAN BE USED ON ADJACENT CARS

INITIALIZATION
11
MODE .NE. 1   TRUE / FALSE
NOTE 12
BEGIN DO LOOP 29 I = 2, N, 2
C2.14--->
29   13
WGT(I) = WGT(I)*WFACTR
14
END OF DO LOOP?   NO / YES
. 2 .
. 13 .
28   15
XW = 0.0
VW = VWALL
T = TO

NOTE 16
BEGIN DO LOOP 32 I = 1, N
17
FC(I) = 0.0
XC(I) = 0.0
VC(I) = VTRN
AC(I) = ACO*G
XP(I) = 0.0
VP(I) = VTRN
32   18
AP(I) = 0.0
19
END OF DO LOOP?   NO / YES
20
I/O  OUTPUT (I)

BEGIN INTEGRATION LOOP. INTEGRATION BY MODIFIED EULER METHOD
04.08*-->
10   21
XW = VW*T

EVALUATE ACCELERATION AT T
22
B.01  FORCE
NOTE 23
BEGIN DO LOOP 707 I = 1, N
24
ACT = 0.0

25
ABS(VC(I)) .GT. 0.01   FALSE / TRUE
26
ACT = ACO*VC(I) /ABS(VC(I))
27
XDD = (FC(I + 1) - FC(I))*G/WGT(I) + ACT

PROJECT POSITION AND VELOCITY AT T+DT
28
XD = VC(I) + XDD*DT
X = XC(I) + VC(I)*DT + .5*XDD*DT*DT
XC(I) = X
VC(I) = XD
707   29
AC(I) = XDD
30
END OF DO LOOP?   NO / YES
31
T = T + DT
XW = VW*T

EVALUATE ACCELERATION AT T+DT

/ 3.01
```

CHART TITLE - PROCEDURES

```
02.32--->*
         |                01
*---------------------------*
| |                         H
|8|        FORCE            H
|.|                         H
|0|                         H
|1|                         H
*---------------------------*
         |
         |  NOTE  02
* * * * * * * * * * * *
*  BEGIN DO LOOP       *
*  706 I = 1, N        *
* * * * * * * * * * * *
         |
         |                03
*---------------------------*
|        ACT = 0.0          |
*---------------------------*
         |
         *   04
FALSE *                 *
+----* ABS(VC(I))       *
     * .GT. 0.01        *
         *
         |TRUE
         |                05
*---------------------------*
|        ACT =              |
|        ACO*VC(I)          |
|        /ABS(VC(I))        |
*---------------------------*
         |
+------->|
         |                06
*---------------------------*
|  XOD = (FC(I +            |
|     I) -                  |
|  FC(I))*G/WGT(I) +        |
|     ACT                   |
*---------------------------*
         |
REVISE POSITION AND
VELOCITY AT T+DT
         |                07
*---------------------------*
|  XO1 = VC(I) -            |
|   AC(I)*DT                |
|                           |
|  XO = XO1 +               |
|  .5*(AC(I) +              |
|   XOD)*DT                 |
|                           |
|  X1 = XC(I) -             |
|   XO1*DT -                |
|  .5*AC(I)*DT*DT           |
*---------------------------*
         |                08
*---------------------------*
| X = X1 + XO1*DT +         |
|  .25*(AC(I) +             |
|   XOD)*DT*DT              |
|                           |
|   XC(I) = X               |
|                           |
|   VC(I) = XD              |
*---------------------------*
         |
706      |                09
*---------------------------*
|   AC(I) = XDD             |
*---------------------------*
         |
         +-------------+

+------------------------->*
                           |
                           *   10
NO  *                          *
+----*   END OF DO LOOP?      *
                           *
                           |YES
                           |               11
*---------------------------*
|        N1 = 1             |
*---------------------------*
                           |
                           *   12
FALSE *                        *
+----*   MODE .EQ. 1          *
                           *
                           |TRUE
                           |               13
*---------------------------*
|        N1 = 2             |
*---------------------------*
+------------------------->|
INTEGRATE POSITION
AND VELOCITY OF
PASSENGERS IN EACH
CAR
                           |
                           |  NOTE 14
* * * * * * * * * * * *
*  BEGIN DO LOOP       *
*  40 I = 1, N, N1     *
* * * * * * * * * * * *
                           |
04.06--------------------->|
                           |               15
*---------------------------*
|  VDIF = VC(I) -           |
|     VP(I)                 |
|                           |
|   AP(I) = 0.0             |
*---------------------------*
                           |
                           *   16
*                          * FALSE
*  VDIF .NE. 0.0 *--+
                           *            |
                           |TRUE      ....
                                      . 4 .
                                      . 01 .
                                      ....
                           |               17
*---------------------------*
|   AP(I) =                 |
|   APO*VDIF/ABS            |
|   (VDIF)                  |
*---------------------------*
                           |
                         / 4.01
```

CHART TITLE - PROCEDURES

```
 C3.16*-->*
          I          01
*--------------------*
I  VP(I) = VP(I) +   I
I     AP(I)*DT       I
I                    I
I  XP(I) = XP(I) +   I
I    VP(I)*DT +      I
I  .5*AP(I)*DT*DT    I
*--------------------*
          I
          I
CALCULATE RELATIVE
POSITION AND VELOCITY
OF PASSENGER
**
          I
          I          02
*--------------------*
I   XPR = XP(I) -    I
I       XC(I)        I
I                    I
I   VPR = VP(I) -    I
I       VC(I)        I           *-----------04.03*-->*
I                    I           I                    I
I    IS = IMP(I)     I           I             40     *   06
*--------------------*           I                  *   *
          I                      I                *       *
          I                      I              *           *   NO
          *  03                  I            *  END OF DO    *--*
        *   *                    I              *  LOOP?    *    I
      *       *                  I                *       *      I
    *           *  TRUE          I                  *   *        I
  *   IS .GE. 7   *--------------*                    *         ....
    *           *                A                    IYES     . 3 .
      *       *                  I                    I        . 15 .
        *   *                    I                    I         ....
          *                      I                    I
          IFALSE                 I                    I
          I                      I                    *  07
          I                      I                  *   *
          I                      I                *       *
          I                      I              *           *  TRUE
IF PASSENGER HAS                 I            *   T .GT. TF   *------*
MOVED INPUT DISTANCES            I              *           *        I
OF SPAC RELATIVE TO              I                *       *          I     AT END OF RUN ,
CAR, LOAD RELATIVE               I                  *   *            I     CALCULATE SEVERITY
VELOCITY INTO VIMP               I                    *              I     INDEX
FOR SUBSEQUENT                   I                    IFALSE         I
SEVERITY                         I                    I              *----------------->*
INDEX CALCULATIONS               I                    I                     900   I        11
SPAC MUST BE IN                  I                    I                    *-------------------*
INCREASING ORDER                 I                    I                    II I               H
          I                      I                    I                    II I     SIC       H
          I                      I           OUTPUT IF PRINT               I. I               H
          *  C4                  I           INTERVAL HAS BEEN             I0 I               H
        *   *                    I           REACHED                       II I               H
      *       *                  I                    I                    *-------------------*
    *           *  TRUE          I                    I                            I
  *   XPR .LT.    *--------------*                    *  08                        I 12
  *   SPAC(IS)  *                A                  *   *                         ...
      *       *                  I                *       *                     .      .
        *   *                    I              *           *  TRUE             . 2.02.
          *                      I            *  TPRNT .GT.   *--*              .      .
          IFALSE                 I              * T + .1*DT *    I                ... 4
          I                      I                *       *      I
          I                      I                  *   *        I
          I                      I                    *         ....
          I                      I                    IFALSE   . 2 .
          I          C5          I                    I        . 21 .
*--------------------*           I                    I         ....
I  IMP(I) = IS + 1   I           I                    I          10
I                    I           I                    I
I VIMP(I,IS) = VPR   I           I                    I          09
*--------------------*           I           *-------------------*
          I                      I           I I                 H
          I                      I           I6 I     OUTPUT     H
          *----------------------*           I. I      (2)       H
                                             I0 I                H
                                             I1 I                H
                                             *-------------------*
                                                      I
                                                      I          10
                                             *-------------------*
                                             I  TPRNT = TPRNT +  I
                                             I       DTP         I
                                             *-------------------*
                                                      I
                                                      I
                                                     ...
                                                   .     .
                                                   . 2.21.
                                                   .     .
                                                     ... 10
```

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
COMMON IMP(20),WGT(20),FC(21),XC(20),VC(20),AC(20),XP(20),VP(20),
       AP(20),VIMP(20,6),DVP(20),DP(20),DM(20),FM(20),URATE(2),
       DR(20),DEL(10,2),FCO(10,2),IF(20),TITLE(20),TO,NCARS,TF,
       WEIGHT,DT,VW,DTP,VTRN,MODE,ACO,APO,SPAC(6),CR(5),WFACTR,
       XW    ,T,G,N,AIMP(20,6,5),SI(20,6,5)
```

CHART TITLE - SUBROUTINE OUTPUT(INO)

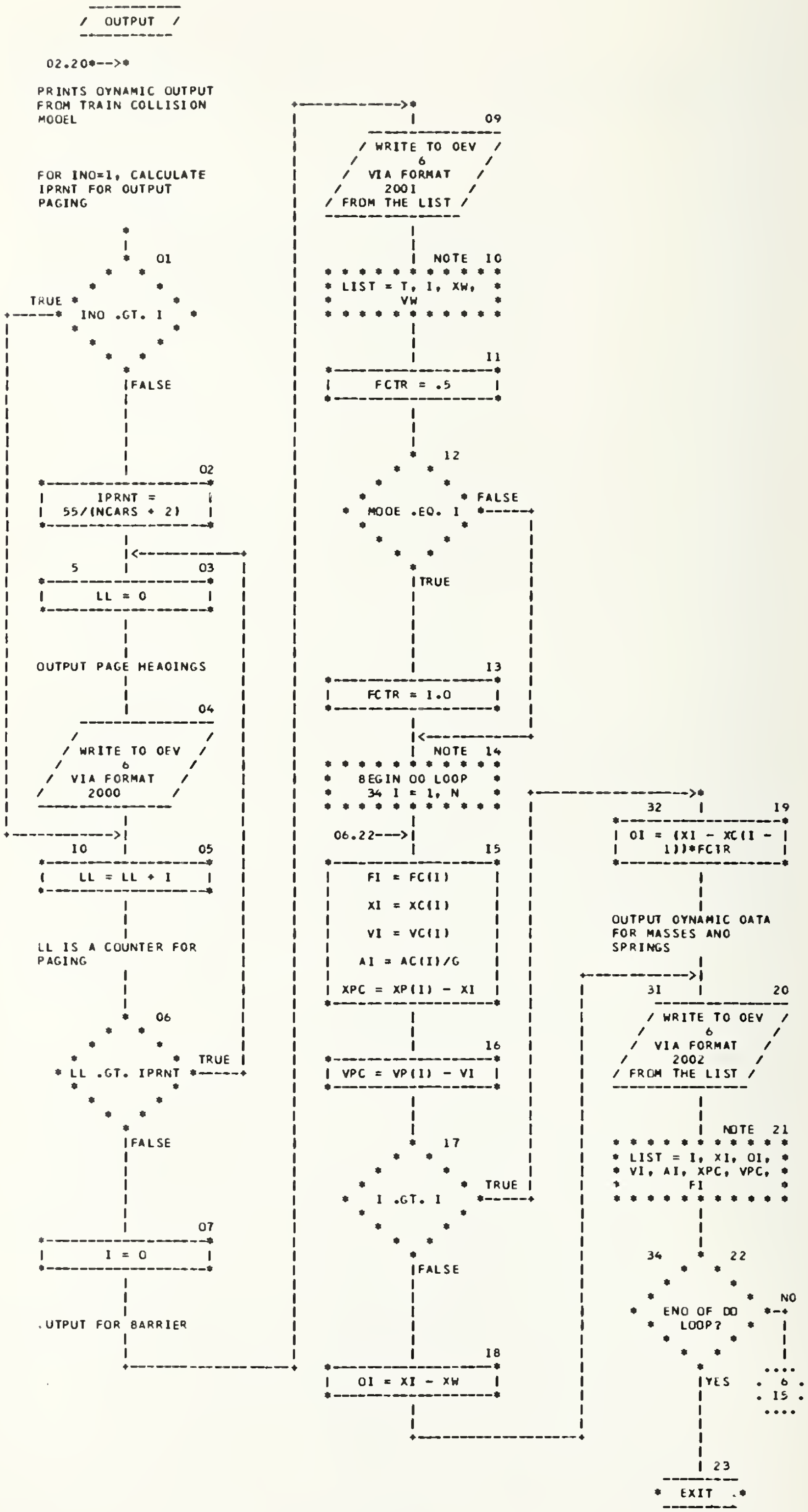


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
         COMMON IMP(20),WGT(20),FC(21),XC(20),VC(20),AC(20),XP(20),VP(20),
               AP(20),VIMP(20,6),DVP(20),OP(20),OM(20),FM(20),URATE(2),
               OR(20),OEL(10,2),FCO(10,2),IF(20),TITLE(20),TO,NCARS,TF,
               WEIGHT,OT,VW,OTP,VTRN,MOOE,ACO,APO,SPAC(6),CR(5),WFACTR,
               XW   ,T,G,N,AIMP(20,6,5),SI(20,6,5)
2000     FORMAT(1H1,1X,4HTIME,6X,3HCAR,5X,2HXC,8X,4HDEFL,6X,2HVC,
              8X,2HAC,8X,3HXPC,7X,3HVPC,7X,1HF   /
              2X,3HSEC,7X,3HNO.,5X,3HFT.,7X,3HFT.,7X,6HFT/SEC,4X,3HG'S,
              7X,3HFT.,7X,6HFT/SEC,4X,3HL8.   //  )
2001     FORMAT(1H0,F8.3,2X,I2,2X,F8.2,12X,F8.2)
2002     FORMAT(1H ,10X,I2,6(2X,F8.2),2X,E10.5)
```

CHART TITLE - SUBROUTINE FORCE

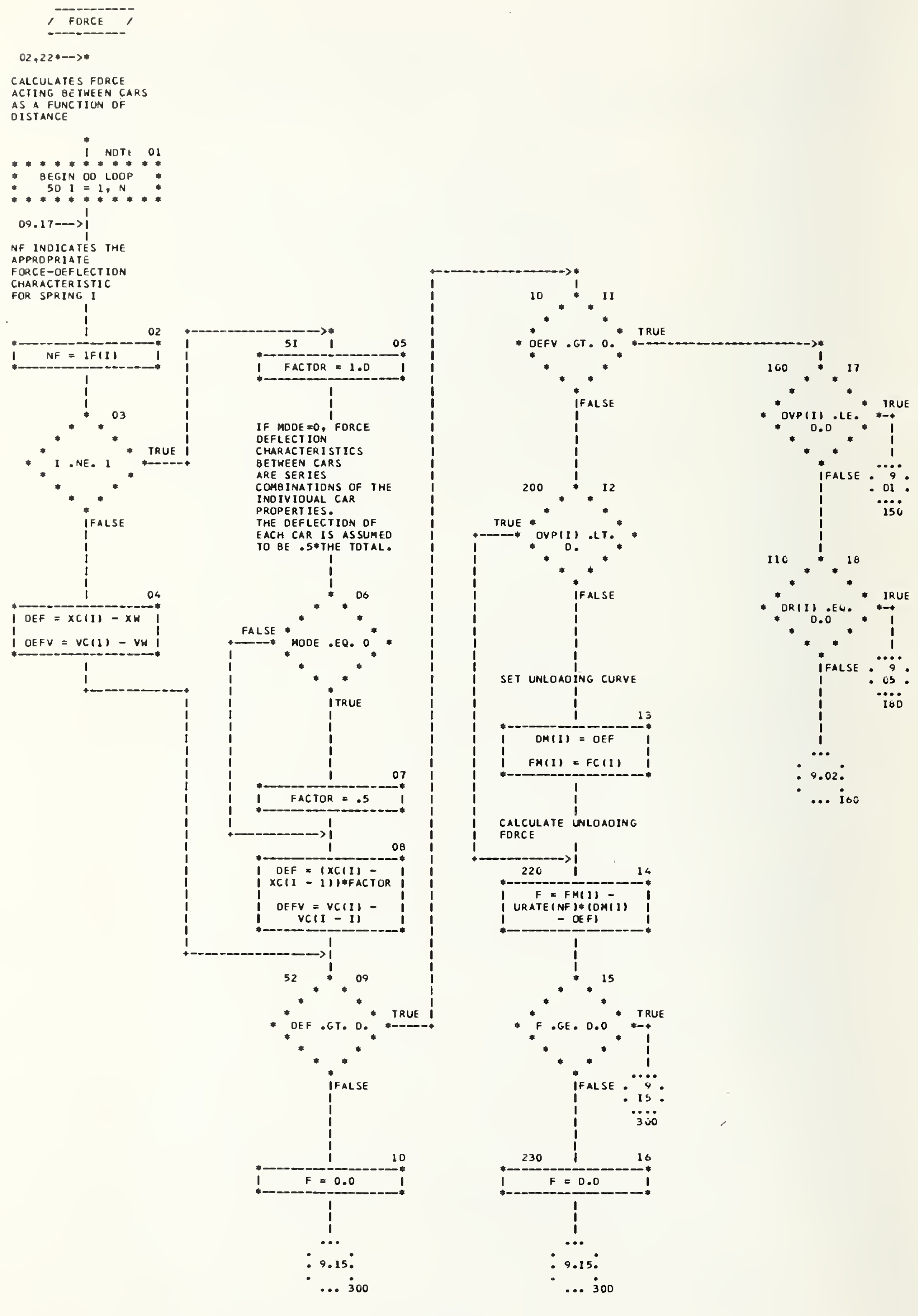


CHART TITLE - SUBROUTINE FORCE

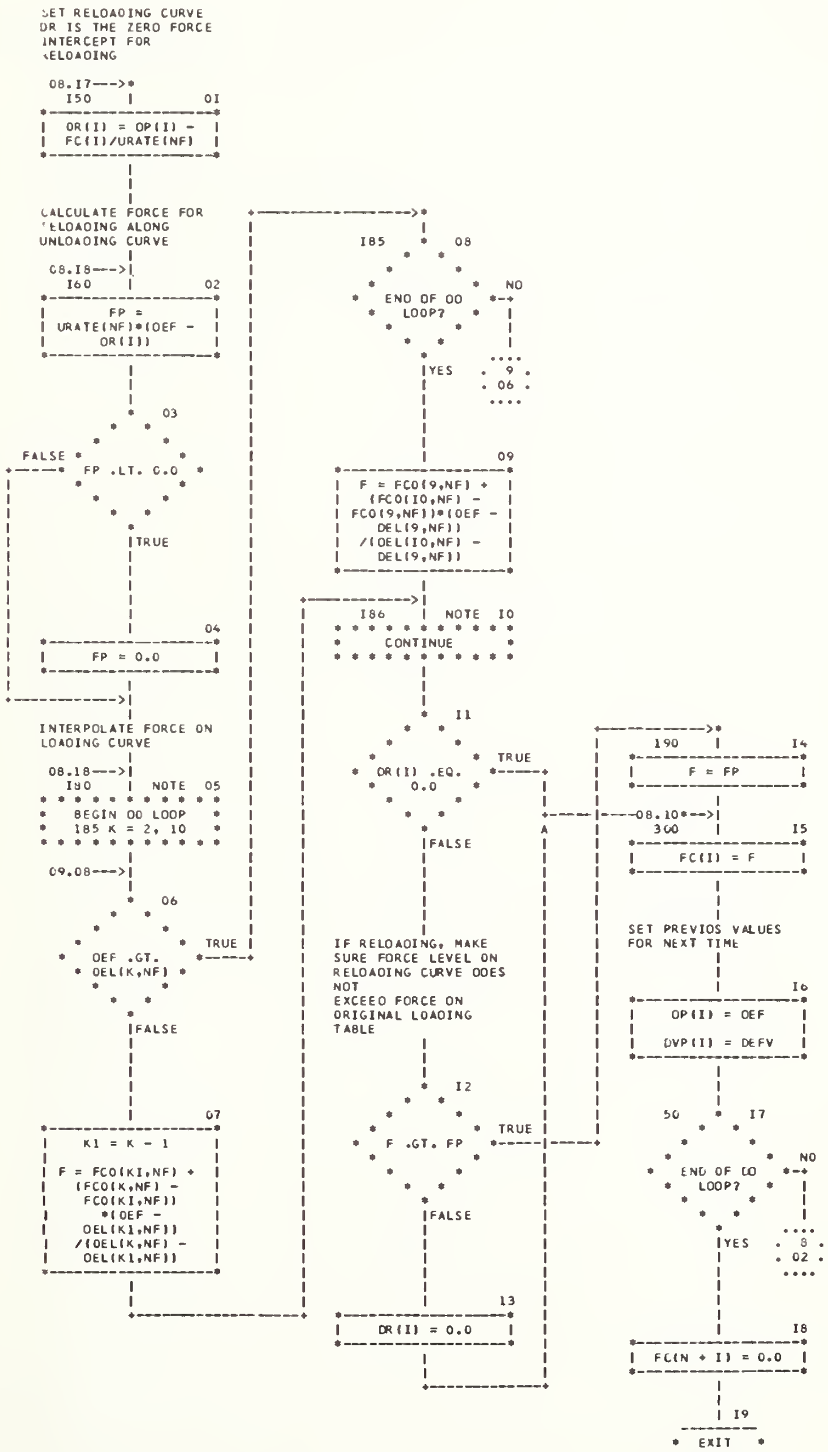


CHART TITLE - NON-PROCEOURAL STATEMENTS

```
COMMON IMP(20),WGT(20),FC(21),XC(20),VC(20),AC(20),XP(20),VP(2C),
       AP(20),VIMP(20,6),DVP(20),OP(20),OM(20),FM(20),URATE(2),
       OR(20),OEL(10,2),FCO(10,2),IF(20),TITLE(20),TO,NCARS,TF,
       WEIGHT,OT,VW,OTP,VTRN,MOOE,ACO,APO,SPAC(6),CR(5),WFACTR,
       XW   ,T,G,N,AIMP(20,6,5),SI(20,6,5)
```

CHART TITLE - SUBROUTINE SIC

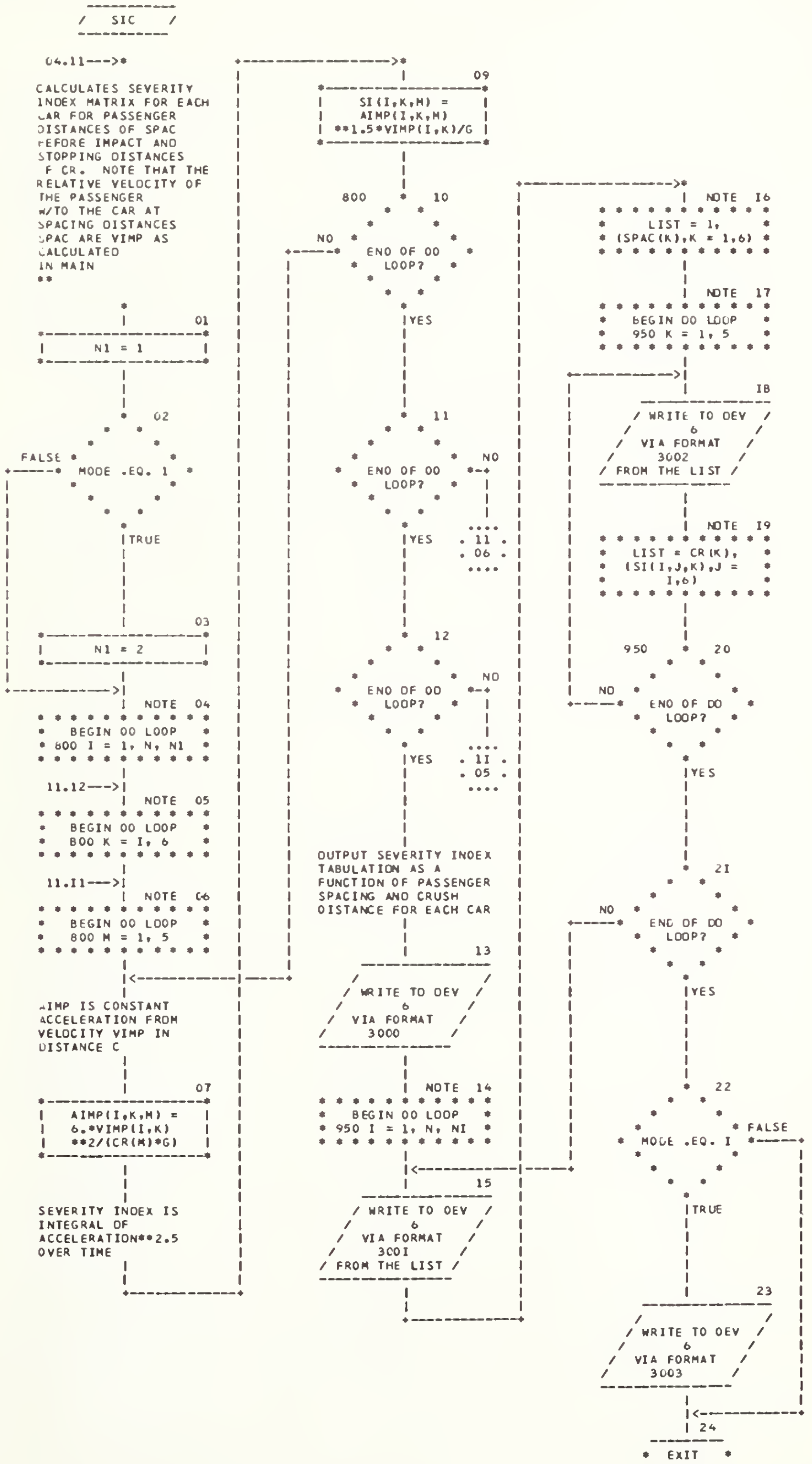


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
         COMMON IMP(20),WGT(20),FC(21),XC(20),VC(20),AC(20),XP(20),VP(20),
                AP(20),VIMP(20,6),DVP(20),OP(20),OM(20),FM(20),URATE(2),
                OR(20),OEL(10,2),FCO(10,2),IF(20),TITLE(20),TO,NCARS,TF,
                WEIGHT,OT,VW,OTP,VTRN,MODE,ACO,APO,SPAC(6),CR(5),WFACTR,
                XW   ,T,G,N,AIMP(20,6,5),SI(20,6,5)
3000     FORMAT(1H1,// 40X,14HSEVERITY INDEX  )
3001     FORMAT(1H0,10X,3HCAR,I3 / 3X,5HCRUSH,22X,12HSPACING - FT    /
                 4X,3HIN.,3X,6F10.2  )
3002     FORMAT(1H0,F7.2,2X,6F10.1  / (1X,F7.2,2X,6F10.1) )
3003     FORMAT(1H0,//47H NOTE: MODE=1, THEREFORE PASSENGERS ASSOCIATED
              20HWITH EVERY OTHER CAR      )
```

CHART TITLE - SUBROUTINE INPUT

```
/ INPUT /

02.05--->*

READS TRAIN COLLISION
MODEL INPUT FROM
CARDS AND PRINTS
INPUT

READ INPUT

RUN TITLE CARD, 80
CHARACTERS

01
/ READ FROM DEV /
/ 5 /
/ VIA FORMAT /
/ 99 /
/ INTO THE LIST /

NOTE 02
* LIST = *
* (TITLE(I),I = *
* 1,20) *

PROGRAM CONTROL

03
/ READ FROM DEV /
/ 5 /
/ VIA FORMAT /
/ 98 /
/ INTO THE LIST /

NOTE 04
* LIST = TO, TF, *
* DT, DTP, MODE *

05
/ READ FROM DEV /
/ 5 /
/ VIA FORMAT /
/ 97 /
/ INTO THE LIST /

NOTE 06
* LIST = NCARS, *
* WEIGHT, VW, VTRN, *
* ACD, APD *

07
/ READ FROM DEV /
/ 5 /
/ VIA FORMAT /
/ 96 /
/ INTO THE LIST /

NOTE 08
* LIST = *
* (SPAC(I),I = 1,6) *

09
/ READ FROM DEV /
/ 5 /
/ VIA FORMAT /
/ 96 /
/ INTO THE LIST /

NOTE 10
* LIST = (CR(I),I = *
* 1,5) *

EACH FCD AND DEL
REQUIRES 2 CARDS WITH
2 VALUES ON SECOND
CARD
BLANK CARD MUST BE
FURNISHED IF NO DATA
FOR LAST TWO ENTRIES.

11
/ READ FROM DEV /
/ 5 /
/ VIA FORMAT /
/ 95 /
/ INTO THE LIST /

NOTE 12
* LIST = *
* (FCD(I,1),I = *
* 1,10) *

13
/ READ FROM DEV /
/ 5 /
/ VIA FORMAT /
/ 95 /
/ INTO THE LIST /

NOTE 14
* LIST = *
* (FCD(I,2),I = *
* 1,10) *

15
/ READ FROM DEV /
/ 5 /
/ VIA FORMAT /
/ 95 /
/ INTO THE LIST /

NOTE 16
* LIST = *
* (DEL(I,1),I = *
* 1,10) *

17
/ READ FROM DEV /
/ 5 /
/ VIA FORMAT /
/ 95 /
/ INTO THE LIST /

NOTE 18
* LIST = *
* (DEL(I,2),I = *
* 1,10) *

19
/ READ FROM DEV /
/ 5 /
/ VIA FORMAT /
/ 95 /
/ INTO THE LIST /

NOTE 20
* LIST = URATE(1), *
* URATE(2) *

IF INDICATES WHICH
FCD-DEL TABLE USED
FOR EACH SPRING.
MUST
BE 1 OR 2

21
/ READ FROM DEV /
/ 5 /
/ VIA FORMAT /
/ 94 /
/ INTO THE LIST /

NOTE 22
* LIST = (IF(I),I = *
* 1,NCARS) *

WFACTR READ ONLY IF
MODE = 1

23
FALSE * MODE .EQ. 1 *
|TRUE

24
/ READ FROM DEV /
/ 5 /
/ VIA FORMAT /
/ 95 /
/ INTO THE LIST /

NOTE 25
* LIST = WFACTR *

+------------->|

PRINT INPUT

/14.01
```

CHART TITLE - SUBROUTINE INPUT

```
   I3.26--->*
            I         01
        ---------------
       / WRITE TO DEV  /
      /       6       /
     /  VIA FORMAT   /
    /     1000      /
   / FROM THE LIST /
   ---------------
            I
            I
            I  NOTE  02
   * * * * * * * * * * *
   *     LIST =        *
   *  (TITLE(I),I =    *
   *      1,20)        *
   * * * * * * * * * * *
            I
            I
            I         03
        ---------------
       / WRITE TO DEV  /
      /       6       /
     /  VIA FORMAT   /
    /     1001      /
   / FROM THE LIST /
   ---------------
            I
            I
            I  NOTE  04
   * * * * * * * * * * *
   * LIST = TO, NCARS, *
   *  TF, WEIGHT, OT,  *
   *  VW, OTP, VTRN,   *
   *  MODE, ACO, APO   *
   * * * * * * * * * * *
            I
            I
            I         05
        ---------------
       / WRITE TO DEV  /
      /       6       /
     /  VIA FORMAT   /
    /     1002      /
   / FROM THE LIST /
   ---------------
            I
            I
            I  NOTE  06
   * * * * * * * * * * *
   *     LIST =        *
   * (FCO(I,1),OEL(I,  *
   *  1),FCO(I,2),     *
   *  OEL(I,2),I =     *
   *     1,10)         *
   * * * * * * * * * * *
            I
            I
            I         07
        ---------------
       / WRITE TO DEV  /
      /       6       /
     /  VIA FORMAT   /
    /     1003      /
   / FROM THE LIST /
   ---------------
            I
            I
            I  NOTE  08
   * * * * * * * * * * *
   * LIST = URATE(1),  *
   *    URATE(2)       *
   * * * * * * * * * * *
            I
            I
            I         09
        ---------------
       / WRITE TO DEV  /
      /       6       /
     /  VIA FORMAT   /
    /     1004      /
   / FROM THE LIST /
   ---------------
            I
            I
            I  NOTE  10
   * * * * * * * * * * *
   *   LIST = (I,I =   *
   *     1,NCARS)      *
   * * * * * * * * * * *
            I
            I
            +------------> (to 11)

   +----------------->*
                      I           11
              ---------------
             / WRITE TO DEV  /
            /       6       /
           /  VIA FORMAT   /
          /     1005      /
         / FROM THE LIST /
         ---------------
                  I
                  I
                  I  NOTE  12
         * * * * * * * * * * *
         * LIST = (IF(I),I = *
         *    1,NCARS)       *
         * * * * * * * * * * *
                  I
                  I
                  I           13
              ---------------
             / WRITE TO DEV  /
            /       6       /
           /  VIA FORMAT   /
          /     1006      /
         / FROM THE LIST /
         ---------------
                  I
                  I
                  I  NOTE  14
         * * * * * * * * * * *
         *     LIST =        *
         *  (SPAC(I),CR(I),  *
         * I = 1,5), SPAC(6) *
         * * * * * * * * * * *
                  I
                  I
                  *     15
                *   *
              *       *
     FALSE  *           *
   +------*  MODE .EQ. 1  *
   I        *           *
   I          *       *
   I            *   *
   I              *
   I              ITRUE
   I              I
   I              I
   I              I           16
   I          ---------------
   I         / WRITE TO DEV  /
   I        /       6       /
   I       /  VIA FORMAT   /
   I      /     1007      /
   I     / FROM THE LIST /
   I     ---------------
   I              I
   I              I
   I              I  NOTE  17
   I     * * * * * * * * * * *
   I     *  LIST = WFACTR    *
   I     * * * * * * * * * * *
   I              I
   +------------->I
                  I 18
              ---------
              * EXIT  *
              ---------
```

CHART TITLE - NON-PROCEOURAL STATEMENTS

```
         COMMON IMP(20),WGT(20),FC(21),XC(20),VC(20),AC(20),XP(20),VP(20),
                AP(20),VIMP(20,6),DVP(20),OP(20),OM(20),FM(20),URATE(2),
                OR(20),OEL(10,2),FCO(IO,2),IF(20),TITLE(20),TO,NCARS,TF,
                WEIGHT,OT,VW,OTP,VTRN,MOOE,ACO,APO,SPAC(6),CR(5),WFACTR,
                XW    ,T,G,N,AIMP(20,6,5),SI(20,6,5)
99       FORMAT(20A4)
98       FORMAT(4F10.0,I5)
97       FORMAT(I5,5X,5FI0.0)
96       FORMAT(6F10.0)
95       FORMAT(BGI0.0)
94       FORMAT(20I4)
1000     FORMAT(1HI,40X,2IHTRAIN COLLISION MOOEL   /  20X,20A4)
1007     FORMAT(1HO,  // 9H WFACTR =,FI0.4)
1001     FORMAT(1HO,25X,4HTO =,F8.4,4H SEC,8X,13HNO. OF CARS =,I4, /
            26X,4HTF =,F8.4,4H SEC,9X,1IHCARWEIGHT =,F12.I,4H LBS /
            26X,4HOT =,F8.5,4H SEC,3X,I8H8ARRIER VELOCITY =,F12.2,4H MPH/
            20X,IOHOT PRINT =,F8.4,4H SEC,5X,16HTRAIN VELOCITY =,FI2.2,
            4H MPH /
           24X,6HMOOE =,14,I5X,14HTRAIN OECEL. =,F12.2,4H G'S  /
            45X,I8HPASSENGER OECEL. =,FI2.3,4H G'S   )
1002     FORMAT(1HO, // IOX,38HTRAIN FORCE-OEFLECTION CHARACTERISTICS   /
            13X,6HFCO(I),IOX,6HOEL(1),IOX,6HFCO(2),IOX,6HOEL(2)   /
            I4X,3HL8S,I3X,3HFT.,I3X,3HLBS,13X,3HFT.  //
            (IOX,F10.I,8X,F7.2,7X,F10.I,8X,F7.2)   )
1003     FORMAT(1HO,10X,26HUNLOADING RATES: URATE(1)=,FI4.2,5HL8/FT,
            10X,9HURATE(2)=,FI4.2,5HLB/FT  /)
1004     FORMAT(1HO, // 5X,7HCAR NO.,28X,15I5 )
1005     FORMAT(IH ,4X,35HFORCE-DEFLECTION CHARACTERISTIC NO.  ,15I5  )
1006     FORMAT(IHO, // IOX,I7HPASSENGER SPACING,13X,I6HPASSENGER OECEL. /
            10X,13HOISTANCE - FT,17X,13HOISTANCE - IN  /
            (13X,F8.2,22X,F8.2)   )
```

CHART TITLE - SUBROUTINE CLEAR(A,B)

```
                   -----------
                  / CLEAR   /
                  ----------

                02.01--->*

               CLEARS (SETS TO ZERO)
               A BLOCK OF STORAGE
               IDENTIFIED BY
               THE ADDRESSES OF THE
               TWO ARGUMENTS
               CALL CLEAR(P,Q) WILL
               CAUSE ALL BYTES TO BE
               SET TO
               ZERO FROM ADDRESS P
               THROUGH THE FULL-WORD
               ADDRESS AT Q

                         *
                         I          01
               *--------------------*
               I    B(1) = 1.0      I
               I                    I
               I       1 = 0        I
               *--------------------*
                         I
                         I<-------------------*
                         I                    I
                  10     *     02             I
                       *   *                  I
                     *       *                I
                   *           * FALSE        I
                 * B(1) .EQ. 0.0 *------------I-------------->*
                   *           *              I               I         04
                     *       *                I     *--------------------*
                       *   *                  I     I     1 = 1 + 1      I
                         *                    I     I                    I
                         ITRUE                I     I    A(1) = 0.0      I
                         I                    I     *--------------------*
                         I                    I               I
                         I                    I               I
                         I                    *---------------*
                         I
                         I 03
                    ----------
                   *  EXIT   *
                    ----------
```

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
DIMENSION A(1),B(1)
```